

AÑO I N.º 4

MICROHOBBY

AMSTRAD

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES AMSTRAD

Semanal

AÑO I N.º 4

150 Ptas.

Canarias 160 ptas.



8.ª FERIA MUNDIAL DEL ORDENADOR PERSONAL

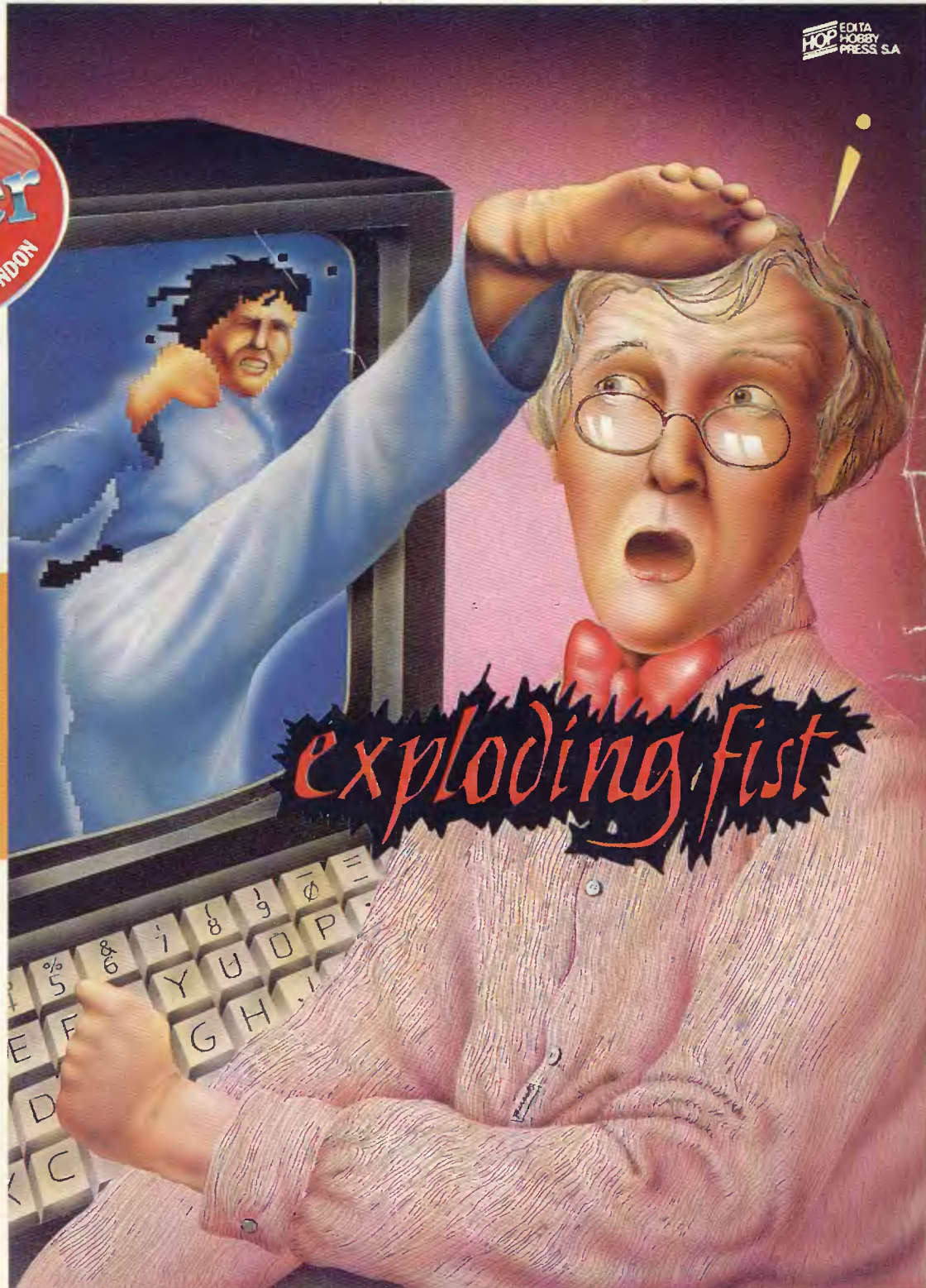
**Microscript,
Procesador CP/M**

**CHARGEN:
HERRAMIENTA
INDISPENSABLE
PARA EL DISEÑO
DE TUS GRAFICOS**

**ARTE EN ALTA
RESOLUCION**

SOFTWARE

**EXPLOIDING
FIST: EL KARATE
DE LA MANO DE
MELBOURNE
HOUSE**



MICROMANIA. Sólo para adictos

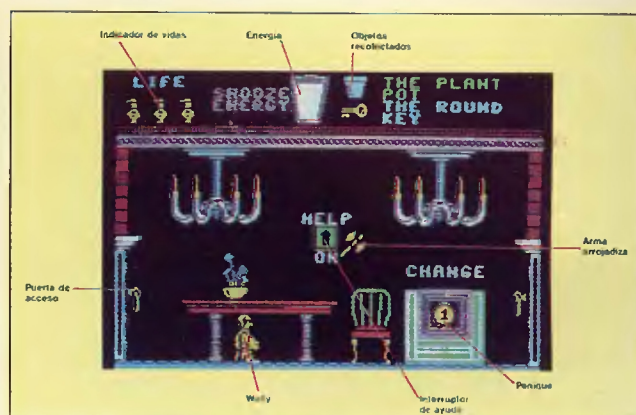


«COMO SE PROGRAMA UN JUEGO»

... para aprovechar a
tope tus posibilidades
como programador.



«AMSTRAD MUSICAL»...
amplios artículos a tu medida.



«PATAS ARRIBA»... la sección que
destripa los mejores juegos, POKE A
POKE, para hacerte invencible.

Una revista con
marcha para los
que necesitan
saber TODO
sobre
ordenadores.

MICRO Mania

HOBBY PRESS S.A.

Año 1 - N° 4

Sólo para adictos

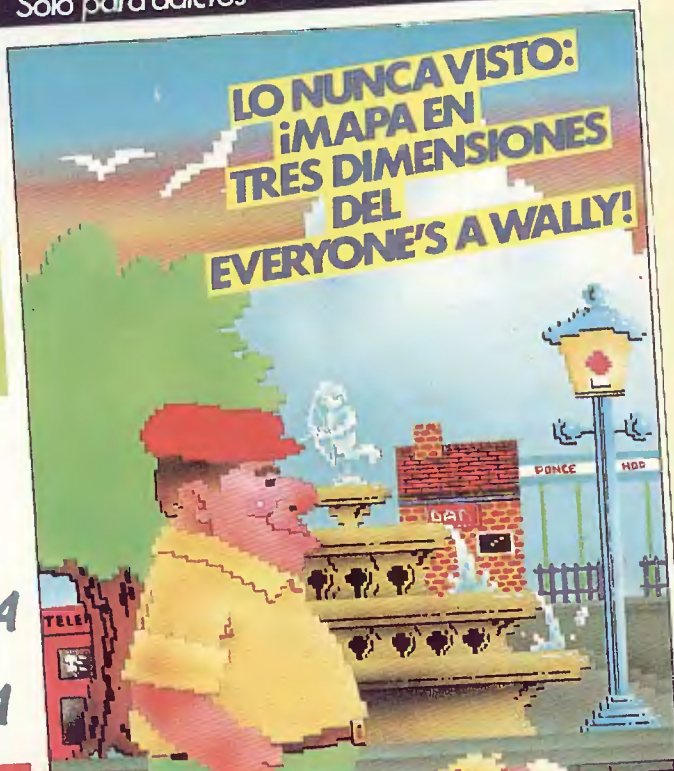
250 Ptas.

¡MUSICA MAESTRO!

TODA
UNA ORQUESTA
EN TU
AMSTRAD

PATAS ARRIBA
LOS "POKES"
DE
PYJAMARAMA
Y AUTOMANIA

LO NUNCA VISTO:
¡MAPA EN
TRES DIMENSIONES
DEL
EVERYONE'S A WALLY!



HOBBY PRESS, S.A. Editamos para gente inquieta.

MICRO

Franquen
Postal



«AMISTRAD MUSICAL»...
amplios artículos a tu medida.

HOBBY PRESS, S.A.

Apartado de Correos

n.º 54.062 (Apartados Altos)

MADRID

Una revista con
marcha para los
que necesitan
saber TODO
sobre
ordenadores.

MICRO Mania

HOBBY PRESS S.A.

Año I · N.º 4

Sólo para adictos

250 Ptas.



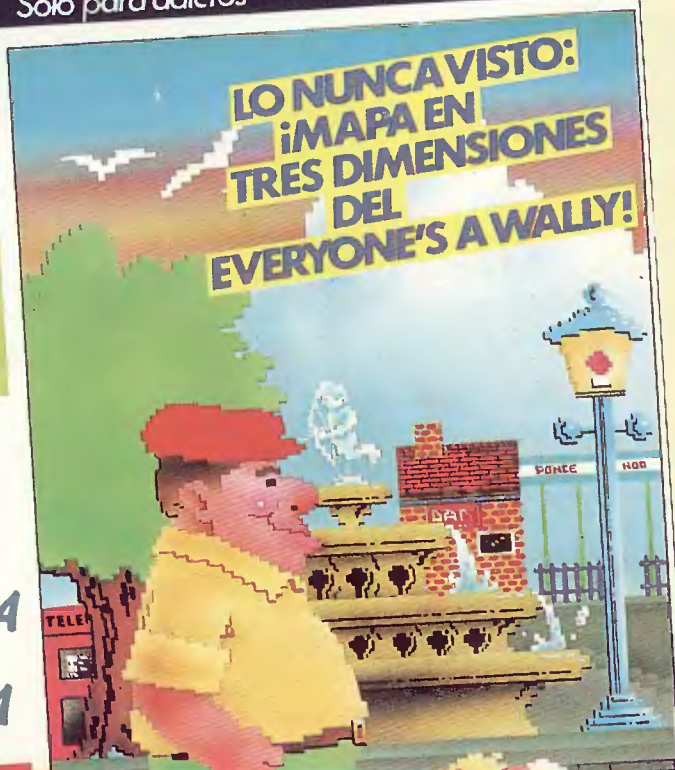
... Y además, la
posibilidad de ganar
una POLAROID si
encuentras al
travieso BYTE
enmascarado.

¡MUSICA MAESTRO!

TODA
UNA ORQUESTA
EN TU
AMSTRAD

PATAS ARRIBA
LOS "POKES"
DE
PYJAMARAMA
Y AUTOMANIA

LO NUNCA VISTO:
¡MAPA EN
TRES DIMENSIONES
DEL
EVERYONE'S A WALLY!



HOBBY PRESS, S.A. Editamos para gente inquieta.

Comer por esta línea

¡¡SUSCRIBETE A Microhobby AMSTRAD Y AHORRA 1.600 PTAS.!!

TARJETA DE SUSCRIPCION AHORRO/REGALO

(Si lo prefieres suscríbete por teléfono (91) 733 50 12 (91) 733 50 16)

Oferta especial para recibir en tu domicilio todo un año la revista semanal *Microhobby AMSTRAD* con un descuento de 1.600 ptas., y el regalo de una cinta original por valor de 2.100 ptas.!

FECHA LIMITE DE RESPUESTA:
30 DE NOVIEMBRE DE 1985
¡RESPONDE HOY MISMO!

¡SEÑALA EL NOMBRE DE LA CINTA QUE
PREFIERES Y LA RECIBIRAS EN TU CASA,
GRATIS, A VUELTA DE CORREO!

Deseo suscribirme a **Microhobby AMSTRAD** durante un año por sólo **5.900 ptas.**, lo que me supone un ahorro de **1.600 ptas.**

El primer número que deseo recibir es el _____

Envíame **GRATIS** la cinta de programas que le indico con una (X)

☐ Beach Head

☐ Pole Position

☐ D.T. Decathlon

NOMBRE _____ EDAD _____

APELLIDOS _____

DOMICILIO _____

CIUDAD _____ PROVINCIA _____

C. POSTAL _____ TELEFONO _____ PROFESION _____

Marco con una (X) en el casillero correspondiente la forma de pago que más me conviene.

☐ Talón bancario adjunto a nombre HOBBY PRESS, S. A.

☐ Giro Postal N.º _____

☐ Contra reembolso del

primer número, junto a la cassette regalo.

☐ VISA N.º _____

☐ MASTER CHARGE N.º _____

Fecha de caducidad de la tarjeta _____

Firma: _____



igo
uina

es sin duda alguna uno
l microprocesador del
tante conocer cómo
ten realizarse con él.
ana intentamos
tarea, sin
unidad.

Consejero Delegado
José I. Gómez-Centurió

Jefe de Publicidad

Concha Gutiérrez

Publicidad Barcelona

José Galán Cortés

Tel: (93) 303 10 22/313 71 62

Secretaria de Dirección

Marisa Cogorro

Suscripciones

M.º Rosa González

M.º del Mar Calzada

**Redacción,
Administración
y Publicidad**

La Granja, s/n
Polígono Industrial de
Alcobendas

Tel.: 654 32 11

Telex: 49 480 HOPR

Dto. Circulación

Carlos Peropadre

Distribución

Coedis, S. A. Valencia, 245
Barcelona

Imprime

ROTEDEC, S. A. Crta. de Irún,
Km. 12,450 (MADRID)

Fotocomposición

Novacomp, S.A.
Nicolás Morales, 38-40

Fotomecánica

GROF
Ezequiel Solana, 16

Depósito Legal:

M-28468-1985

Derechos exclusivos

de la revista

**COMPUTING with
the AMSTRAD.**

Representante para Argentina, Chile,
Uruguay y Paraguay, Cia.
Americana de Ediciones, S.R.L. Sud
América 1.532. Tel.: 21 24 64. 1209
BUENOS AIRES (Argentina).

M. H. AMSTRAD no se hace
necesariamente solidario de las
opiniones vertidas por sus
colaboradores en los artículos
firmados. Reservados todos los
derechos.

Se solicitará control OJD



Todo el mundo ha estado ahí, mostrando sus
últimas novedades y sus próximos proyectos, en
la medida que la competencia lo permite.

AMSTRAD semanal se ha trasladado a
Londres para dar cumplida información del
acontecimiento.

8 Banco de pruebas

MICROSCRIPT, un procesador de textos
profesional, que ofrece gran potencia y
versatilidad dentro de la línea del software
CP/M.

Capaz de manejar, con dos unidades de
disco, un texto de una longitud superior a las
26.000 palabras.

Serie oro 14

Los aficionados a programarse sus propios
juegos, o aquéllos que deseen fervientemente
añadir originales a sus creaciones, encontrarán
indispensable esta utilidad, pensada para
diseñar a voluntad juegos de caracteres en el
Amstrad, ejemplo en sí mismo de belleza y
estilo.

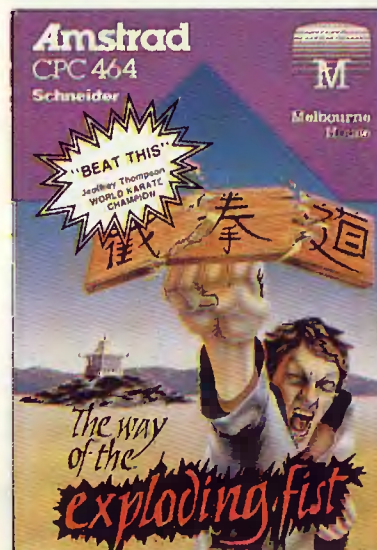
ProgramAcción 28

Tal vez una de las cualidades más fascinantes
de los ordenadores es su capacidad gráfica, la
posibilidad de plasmar en la pantalla hermosas
imágenes.

El **Amstrad** posee gran cantidad de
comandos gráficos, que nos permiten dibujar en
alta y baja resolución. Explicamos cómo y por
qué funcionan, y, además demostramos qué
fácil es conseguir con **Amstrad Basic** el
famoso efecto de ZOOM.

18 Mr. Joystick

Melbourne House, en su línea de productos
de gran calidad, nos sorprende con un
magnífico programa, el **SXPLODING FIST**,
en el que nos convertimos en un afamado
campeón de la Artes Marciales, el cual debe
superar una serie de pruebas y vencer a unos
cuantos enemigos en combate para probar su
habilidad y valor.



Director Editorial
José I. Gómez-Centurión

Director Ejecutivo
Victor Prieto

Subdirector
José María Díaz

Redactora Jefe
María García

Diseño
José Flores

Colaboradores
Francisco Portalo
Pedro Sudón

Miguel Sepúlveda
Francisco Martín

Jesús Alonso

Secretaría Redacción
Carmen Santamaría

Fotografía
Carlos Candel

Javier Martínez

Portada

Manuel Barco

Ilustradores

J. Igual, J. Pons, F. L. Frontán,
J. Septien, Pejo, J. J. Mora

Edita

HOBBY PRESS S.A.

Presidente

María Andriño

Consejero Delegado

José I. Gómez-Centurión

Jefe de Publicidad

Concha Gutiérrez

Publicidad Barcelona

José Galán Cortés

Tel: (93) 303 10 22/313 71 62

Secretaría de Dirección

Marisa Cogorro

Suscripciones

M.ª Rosa González

M.ª del Mar Calzada

**Redacción,
Administración
y Publicidad**

La Granja, s/n

Polígono Industrial de

Alcobendas

Tel.: 654 32 11

Telex: 49 480 HOPR

Dto. Circulación

Carlos Peropadre

Distribución

Coedis, S. A. Valencia, 245
Barcelona

Imprime

ROTEDEC, S. A. Crta. de Irún,

Km. 12,450 (MADRID)

Fotocomposición

Novocomp, S.A.

Nicolás Morales, 38-40

Fotomecánica

GROF

Ezequiel Solana, 16

Depósito Legal:

M-28468-1985

Derechos exclusivos

de la revista

**COMPUTING with
the AMSTRAD.**

Representante para Argentina, Chile,
Uruguay y Paraguay, Cia.
Americana de Ediciones, S.R.L. Sud
América 1.532. Tel.: 21 24 64. 1209
BUENOS AIRES (Argentina).

M. H. AMSTRAD no se hace
necesariamente solidaria de las
opiniones vertidas por sus
colaboradores en los artículos
firmados. Reservados todos los
derechos.

Se solicitará control OJD

MICROHOBBY

AMSTRAD

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES AMSTRAD

Semanal

AÑO I N.º 4

Año I • Número 4 • 24 al 30 de septiembre de 1985
150 ptas. (sobretasa Canarias, 10 ptas.)

4 Primera plana

En Londres se ha celebrado una de las ferias informáticas más importantes del mundo: la P. C. WORLD SHOW, la exposición mundial de ordenadores personales.



Todo el mundo ha estado ahí, mostrando sus últimas novedades y sus próximos proyectos, en la medida que la competencia lo permite.

AMSTRAD semanal se ha trasladado a Londres para dar cumplida información del acontecimiento.

8 Banco de pruebas

MICROSCRIPT, un procesador de textos profesional, que ofrece gran potencia y versatilidad dentro de la línea del software CP/M.

Capaz de manejar, con dos unidades de disco, un texto de una longitud superior a las 26.000 palabras.

Serie oro 14

Los aficionados a programarse sus propios juegos, o aquellos que deseen fervientemente añadir originales a sus creaciones, encontrarán indispensable esta utilidad, pensada para diseñar a voluntad juegos de caracteres en el **Amstrad**, ejemplo en sí mismo de belleza y estilo.

22 Código Máquina

El registro acumulador es sin duda alguna uno de los más importantes del microprocesador del **Amstrad**. Es muy importante conocer cómo funciona y qué cosas pueden realizarse con él. En el artículo de esta semana intentamos simplificar al máximo esta tarea, sin precipitaciones y con profundidad.

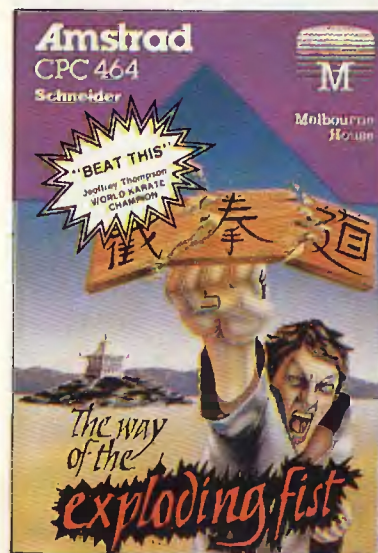
ProgramAcción 28

Tal vez una de las cualidades más fascinantes de los ordenadores es su capacidad gráfica, la posibilidad de plasmar en la pantalla hermosas imágenes.

El **Amstrad** posee gran cantidad de comandos gráficos, que nos permiten dibujar en alta y baja resolución. Explicamos cómo y por qué funcionan, y, además demostramos qué fácil es conseguir con **Amstrad Basic** el famoso efecto de ZOOM.

18 Mr. Joystick

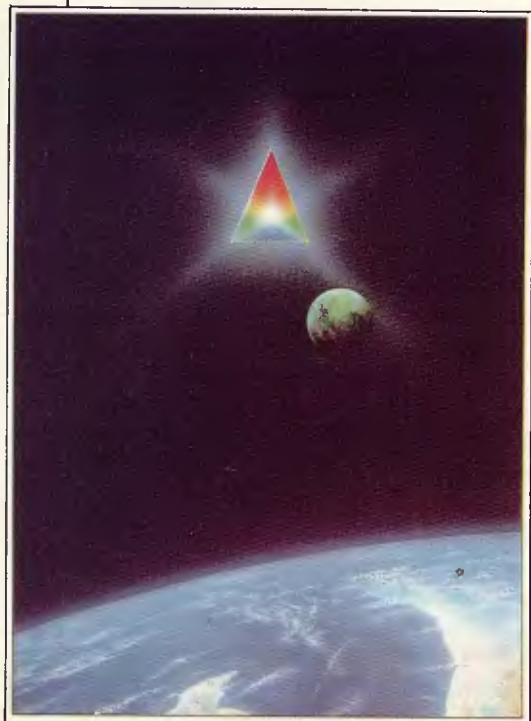
Melbourne House, en su línea de productos de gran calidad, nos sorprende con un magnífico programa, el **SXPLODING FIST**, en el que nos convertimos en un afamado campeón de la Artes Marciales, el cual debe superar una serie de pruebas y vencer a unos cuantos enemigos en combate para probar su habilidad y valor.



Primera plana

LA INFORMATICA INVADE EL OLIMPIA

Londres ha sido la sede de la octava feria mundial del ordenador personal. Realizada durante los días 4-8 de septiembre en el marco incomparable del OLYMPIA.



La muestra de 1985, ha reunido toda la gama de productos que componen el mundo del ordenador personal, desde los ordenadores para negocios, hasta los ordenadores dirigidos al mundo de la educación, pasando por el popular ordenador de juegos.

Esta exposición ha sido la más amplia de las ocho celebradas en su historia habiendo participado en ella más de 200 compañías.

Por primera vez, la muestra ha ocupado las dos salas del OLYMPIA, estando alojada en el OLYMPIA 2, una exclusiva para los ordenadores de gestión, mientras que en el NATIONAL HALL, se encontraban reunidos los demás aspectos de la industria informática.

Como en años anteriores, la feria ha sido el marco de los lanzamientos de las grandes compañías, haciendo su debut en esta ocasión el superconocido ATARI 520 ST, los APRICOT F2 y F10, el procesador de textos **Amstrad** 8256 y la nueva gama de productos PHILIPS.

Los consumidores de software, dispusieron de las últimas creaciones realizadas por las firmas internacionales de software.

MICRO GEN, KONAMI, FIREBIRD, US. GOLD, ARGUS, MIRROR-SOFT, OCEAN, LAMASOFT y muchas otras casas de reconocido prestigio, presentaron sus últimos productos a los visitantes de la muestra.

Cabe destacar como característica más importante del evento, la cruenta guerra de precios que mantienen los fabricantes, con el objetivo de acaparar el mercado del ordenador doméstico-semiprofesional, apto para pequeños negocios.

Intentando por todos los medios ofrecer los productos más potentes y con mayor número de prestaciones al precio más competitivo.

Como máximo exponente de esta política informática, tenemos los modelos ATARI, que además de presentar el supermodelo 520 ST, también han desarrollado software de aplicaciones profesionales y de gestión, utilizando el sistema operativo GEN diseñado por Digital Research para el uso del 520.



COMMODORE, que tampoco quiere quedarse atrás en la carrera por actualizar sus modelos con un precio revolucionario, presentó su COMMODORE 128, con una extensa gama de periféricos.

El 128, viene acompañado con una creciente librería de aplicaciones de 128 K, además de los cientos de programas CP/M, de uso en negocios y en todos los campos de aplicación informática.

SINCLAIR por su parte, continúa con su campaña de promoción del QL, recortando su precio prácticamente a la mitad y presentando una gama creciente de software, desarrollada por PSION y EIDERSOFT, en esfuerzo titánico para no perder el tren del ordenador con serias aplicaciones profesionales.

Entre los productos más sofisticados de esta nueva ofensiva SINCLAIR tenemos:

BASE RESULT, el tercer elemento de la serie de productos de HANDIC SOFTWARE.





DATA PLUS, un paquete de aplicaciones para manejo de datos.

Una hoja de expansión de memoria de la firma SENTINEL SOFTWARE

El sector de la educación ha estado fuertemente representado por ACORN, APRICOT y RESEARCH MACHINES, cuyas creaciones en el campo del *hardware*, dirigido especialmente a colegios y enseñanza, viene apoyado por una extensa serie de software, realizado por la recientemente creada BRITISH EDUCATIONAL SOFTWARE ASSOCIATES CONSORTIUM.

Los ordenadores portátiles también ocuparon su parcela en la feria; IBM AT-COMPATIBLE TAVA FLYER, en su versión de 640 K, el ENCORE PLUS, OSBORNE EXPRESS y el APRICOT PORTATIL FP.

El campo de las comunicaciones y las redes de trabajo, también tuvo su importante representación, entre ellos debemos destacar las participación de los sistemas TOURS ICON NETWORK, que fueron los ganadores del pasado premio concedido por STANDAR MICRO BUSSINES, cuando realizaron su presentación con KGB MICROS.

Otros productos que merecen una destacada mención en el mercado de las redes de comunicación, son los APRICOT NETWORKS y el paquete de comunicaciones TELDECOM GOL de la conocida casa PSION.

Los ordenadores RESEARCH cuentan con dos nuevos productos; la po-

derosa red de trabajo NIMBUS NETWORK, que permite enlazar máquinas de 16 bits y la más modesta CHAIN NETWORK, utilizada sobre todo en colegios para transmisión de datos.

Sin lugar a dudas, la representación más colorista y dinámica de la super feria corrió a cargo de las firmas de software, sus lanzamientos venían acompañados por toda clase de llamadas publicitarias. Música e imagen intentaban atraer al público en una auténtica lucha por el interés.

Karatecas, un doble de Sylvester Stalone en el papel de acorralado, música de los Rollings, videos y demás medios eran el soporte fundamental de los stands.





Todas las casas de software más conocidas tenían una nutrida representación en la muestra, exceptuando por supuesto a los misantrópicos e inefables de **ULTIMATE**, los cuales como única publicidad cuentan con la insuperable calidad de sus productos, y rehusan sistemáticamente el acudir a cualquier exposición.

Las demás casas no escatimaron en medios para presentar sus creaciones, entre las cuales no podemos dejar de citar:

MICRO GEN, con su revolucionario lanzamiento, **THE SHADOW OF THE UNICORN**, un juego para **SPECTRUM** que incluye una cinta y un cartucho de ROM, aumentando de esta manera la memoria del ordenador de 48 a 64 K, totalmente revolucionario en el mundo de este pequeño ordenador.

Para **Amstrad**, este maravilloso juego todavía tardará unos meses en aparecer, pero en cambio sí pudimos contemplar el famoso **EVERYONE'S A WALLY** por fin en versión para **Amstrad**.

OCEAN, tampoco quiere perder el tren de la actualidad y nos dio a conocer sus nuevos programas pa-

ra la creación de juegos dinámicos.

Sus productos; **Amstrad LASER BASIC** y **Amstrad LASER COMPILER**, ponen al alcance de cualquiera la realización de los más vistosos juegos.

Con el **LASER BASIC**, podemos crear directamente desde el basic, cualquier juego de animación que nuestro talento sea capaz de concebir.

Esta poderosa herramienta de la programación, permite utilizar los famosos **SPRITES**, además permite el movimiento simultáneo de varias figuras, superposiciones, transparencias y demás efectos que hacen que los juegos comerciales sean tan magníficos.

Pero la belleza del programa reside en que todo esto se realiza desde el basic, obteniendo una velocidad de movimiento muy buena.

Pero si a pesar de todo, el juego nos resulta un poco lento, todavía tenemos el **LASER COMPILER**, que compilará nuestro programa, haciéndolo mucho más veloz.

MELBOURNE HOUSE, con el stand repleto de karatecas, apoyaba su **EXPLODING FIST**, disponible para los tres ordenadores; **SPECTRUM**, **COMMODORE** y **Amstrad**.

Entre los juegos disponibles para **Amstrad**, debemos hacer obligada mención de:

THE HOBBIT, la versión en ordenador de la mítica obra de **TOLKIEN**.

STARION, una terrible contienda interestelar con magníficos efectos en tres dimensiones.

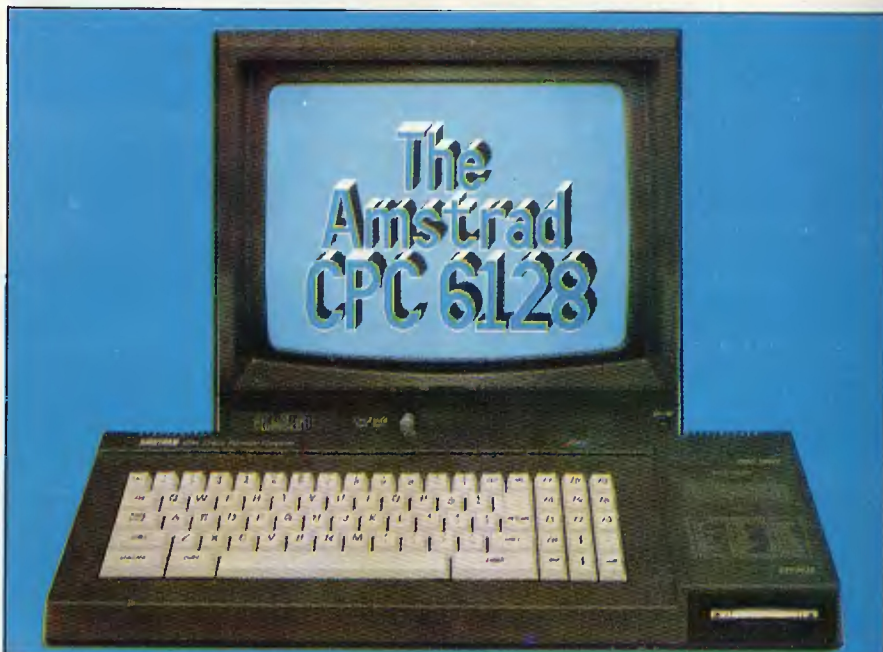
Y la próxima aparición en el mercado de **TERRORMOLINOS**, una aventura en la Costa del Sol.

ADVENTURE INTERNATIONAL, con su última aventura **ROBIN OF SHERWOOD**, más conocido como **ROBIN HOOD**.

Por ahora en inglés, pero según nos confirmó el director de productos de la firma, muy pronto será lanzada en España la versión en castellano (recordemos que **ADVENTURE INTERNATIONAL**, son los creadores de **GREMLINS**).

DOMARK, siempre en su línea de actualidad y adaptación de temas con éxito garantizado, exponía el popular **A VIEW TO A KILL**, junto con su último lanzamiento **CODENAME II**.

ARIOLA SOFT, lanzando su última realización **HARD HAT MACK**. Promocionando también la inminente salida de **ARCHON**, **ONE ON ONE**, **REALM OF IMPOSSIBILITY** y **WIZARD**.



Y sus proyectos en un futuro un poco más lejano:

ARCHON II, THREE DAYS IN CARPATHIA, THINK y SKIFOX, todo un amplísimo panorama para los consumidores de software recreativo en el **Amstrad**.

MIRRORSOFT, presentando su BOULDER DASH, DINAMITE DAN y sus controladores de vuelo con aviones de despegue vertical, VSTOL.

DEEP THOUGH, con su último programa para **Amstrad**, el ajedrez tridimensional, con voz sintetizada, un programa al estilo PSION, que ha realizado el ajedrez más espectacular e impresionante en la historia de los ordenadores, THE QL CHESS, una maravilla de la cual solamente pueden disfrutar los poseedores de un QL, o los afortunados que tengan un MACKINTOSH.

MARTECH, que bajo el lema de THE BATTLE BEGINS (comienza la batalla), promociona su última creación, ZOIDS.

ACTIVISION, presentando el GHOSTBUSTERS y anunciando sus próximos juegos; HACKER, MIND-SHADOW, BALLBLAZER, THE ACTIVISION DECATHLON, ON COURT TENNIS y ON FIELD FOOTBALL.

Por fin en el stand de **Amstrad**, después de visitar toda la magna exposición, cerramos nuestro trayecto a través de la feria en el stand de la conocida marca.

Las estrellas de la firma y que acapaban toda la atención del público que visitaba la feria, se dividía en dos frentes:

El **Amstrad** CPC-6128.

El **Amstrad** PCW-8256.

Dos modelos con una concepción muy distinta, pero con una característica similar; su bajo precio.

Del CPC-6128, poco nos queda que decir, pues fue analizado con profundidad en el anterior número de nuestra revista.

Con un precio en el mercado inglés de 300 libras.

Y la gran novedad de la casa el PCW-8256, un ordenador con mayores pretensiones, que intenta entrar de lleno en el campo profesional.

El nuevo modelo está concebido como un procesador de textos de amplia capacidad de almacenamiento.

El equipo se compone de teclado, monitor de fósforo verde con disco integrado y una impresora.

El teclado consta de 82 teclas, diseñadas para trabajar directamente con el programa de proceso de textos.

Tiene una memoria RAM de 256 K y en cada disco es capaz de almacenar un máximo de 180.000 caracteres, lo que equivale a unas 94 páginas de texto, constituidas cada una por 30 líneas de 64 caracteres cada una.

El monitor en fósforo verde permite un formato de escritura de 90 columnas por 32 filas.

Utiliza un disco integrado en el monitor con una capacidad de 180 K formateadas por cara.

El sistema operativo utilizado es el eficiente CP/M 3.0.



Además del programa para el proceso de textos que es el LOCO SCRIP, podemos utilizar todos los programas disponibles en CP/M 80; BASIC, PASCAL, COBOL, C, FORTRAN, MICROPROLOG, FORTH, etcétera.

Lo que nos abre un amplio espectro de posibilidades de utilización de la máquina; no solamente el proceso de textos.

La impresora es unidireccional, con una escritura de alta calidad a 20 cps y en calidad normal de 90 cps, utilizando hasta 7 tipos distintos de letra.

Además del sistema de tracción para papel continuo, también dispone de rollo de fricción para el tratamiento de hojas separadas de texto.

En definitiva un artefacto francamente interesante para aquéllos que quieran introducirse seriamente en el mundo de la gestión. Y esto por el módico precio de 399 libras.

Esperemos a ver qué ocurre en la próxima feria en el 1986.



MICROSCRIPT, A LA SOMBRA DE CP/M

Anteriormente tuvimos ocasión de analizar un programa concebido alrededor de un concepto central muy claro: la comodidad de uso. Todo para el usuario. Sin embargo, como ya dijimos, AMSTWORD es una aplicación relativamente modesta: concebida para el uso con una sola unidad de disco y para el tratamiento de documentos de pequeña extensión.

M

ICROSCRIPT es otra historia completamente distinta, y, como decir MICROSCRIPT es decir CP/M, vamos a retroceder en el tiempo hasta 1977, a la prehistoria de la informática, aquellos tiempos heroicos en los que los poseedores de una computadora no sabían muy bien que hacer con ella.

El nacimiento de CP/M: Todos para uno

Gary Kildall era uno de éstos, pero en lugar de sentarse a lamentar la falta de «algo» que estableciera un orden en el caos de incompatibilidades que afectaba a los ordenadores, creó el «**CONTROL PROGRAM FOR MICROCOMPUTERS**», Su Majestad CP/M.

Este programa servía de intermediario entre el hardware del sistema (los circuitos) y otras aplicaciones como proceso de textos, bases de datos, etc.

Por fin, los fabricantes de software pudieron dejar de preocuparse para qué ordenador escribían sus productos. Con tal que la máquina a la que iban destinados ejecutaría CP/M, no había que preocuparse de nada más. Un mismo programa correría en docenas de equipos distintos.

El abaratamiento de costes y la compatibilidad revolucionó la informática hasta tal punto, que ahora mismo usted tiene un AMSTRAD en casa capaz de ejecutar MICROSCRIPT a un precio razonable.

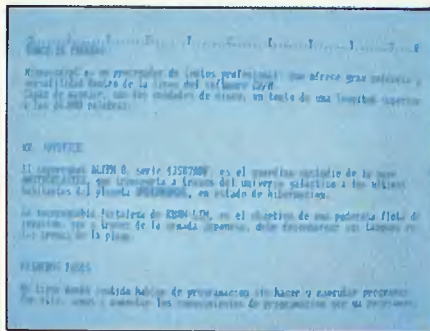
Este programa es un típico producto CP/M, y gira alrededor de un único eje: POTENCIA.

Sin embargo, su estructura responde a los tiempos en que había que aprovechar al máximo la memoria disponible en el ordenador (las memorias eran muy caras), y no había sitio para florituras del tipo de pantallas de ayu-

da; el peso de las consultas descansaba exclusivamente en los manuales.

¿Potencia o comodidad?

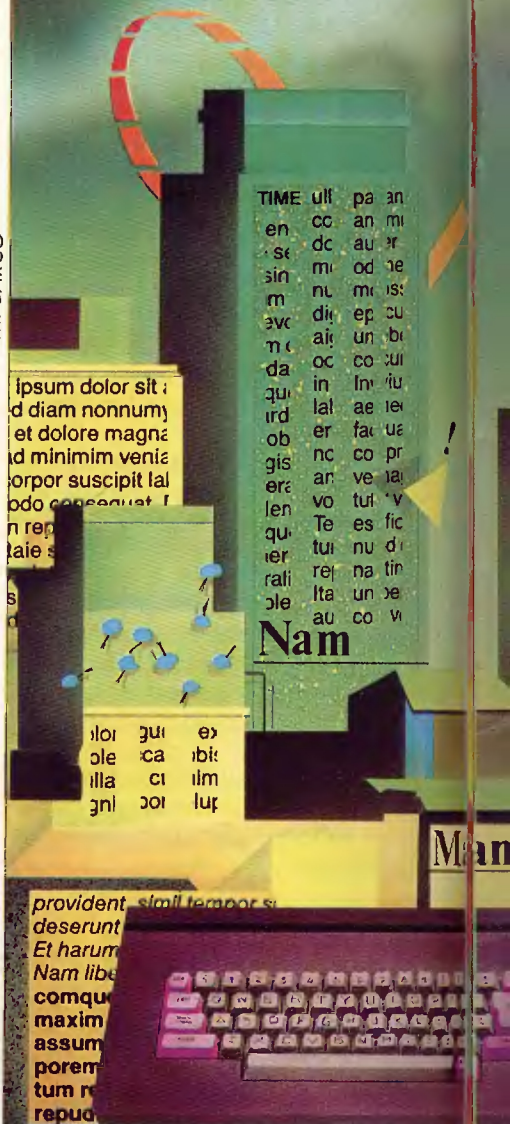
Así, este tipo de programas sacrifican la comodidad de uso a una gran potencia (expresada en la cantidad de funciones disponibles, 57 en MICROSCRIPT), obligando al usuario a consultar continuamente los manuales.



Llegados a este punto, el veredicto que caiga sobre MICROSCRIPT es cuestión de las preferencias y necesidades de cada uno: si su trabajo requiere el manejo de grandes cantidades de texto, o desea la libertad de hacer con él prácticamente todo lo que se imagine, a nivel de formatos de impresión, congregación de documentos desde disco, mailing, etc., este es su programa.

No obstante, tendrá que enfrentarse a un programa difícil de usar, con una «**interfaz de usuario**» deficiente comparada con los procesadores de texto modernos, y unos copiosos manuales, por desgracia mal traducidos, que deberá consultar frecuentemente hasta que domine todos los recursos que MICROSCRIPT pone a su disposición.

M. BARCO



Desde nuestro propio (y subjetivo) punto de vista, MICROSCRIPT ofrece una de cal y otra de arena; el resultado es de tablas.

Esperemos que AMSTRAD, a través de sus distribuidores en España, ofrezca pronto otros programas que mantengan el alto nivel de potencia que retrata a MICROSCRIPT y posean una forma de manejo más sencilla, acorde con las actuales tendencias que dominan en el mundo del software profesional.

La única «**excusa**» que aceptaríamos si estos nuevos programas no aparecieran pronto, sería el aducir que la escasa memoria del AMSTRAD no permite tales libertades.

Y la aceptaríamos a duras penas porque, entre otras cosas, para algo está el disco:

Tomando contacto con MICROSCRIPT

Para arrancar el programa tenemos que introducirnos en el CP/M, tecleando `lcpm + ENTER`.

Cuando aparece el prompt del sistema operativo, la divisa «A», simplemente tecleando «script» aparece el menú principal del programa. Es el siguiente:

1. C Crear un documento
2. E Editar un documento
3. R Reformar un documento
4. S Búsqueda y sustitución
5. P Imprimir un documento
6. F Gestión de ficheros
7. ? Lista índice de documentos
8. * Vuelta al sistema (al CP/M)

Cada una de estas opciones involucra el uso de las funciones del programa, numeradas del 1 al 57, y que se obtienen normalmente mediante secuencias de teclas encabezadas por CTRL o ESC. Por tanto, es difícil hablar de comandos asociados a una determinada opción, pues algunos de ellos son compartidos por varias.

Normalmente, el proceso que se sigue para completar un documento, desde la pantalla a la impresora, implica los pasos 1, 2, 3 y 5 del cuadro anterior. Más tarde volveremos sobre el tema, de momento vamos a centrarnos en los pasos 4 y 6, puesto que los dos restantes son obvios.

OPCION S (BUSQUEDA Y SUSTITUCION)

Como su propio nombre indica, esta opción está pensada para encontrar y/o sustituir determinada cadena de caracteres de un documento, que normalmente se encontrará en el disco (el comando `ESC + F` cumple la misma función con un documento presente en memoria).

Lo primero que se nos pregunta, lógicamente, es el nombre del fichero a inspeccionar, el cual puede encontrarse en cualquiera de las dos unidades de disco. Hay que indicar en cuál, anteponiendo el identificador de unidad al nombre de fichero. Así, «B: CARTA.DOC» le dice al programa que investigue el fichero llamado «CARTA.DOC» localizado en la unidad «B».

Si por cualquier causa nos arrepentimos y no queremos investigar ese documento, pulsando ESC se vuelve al menú principal. Una vez seleccionado el fichero, tenemos que teclear la cadena de caracteres que se pretende encontrar o sustituir.

Acto seguido, el programa inquiere sobre si lo que interesa es:

- a) Encontrar la cadena.
- b) Sustituirla por otra a lo largo de todo el documento.
- c) Contar el número de veces que aparece en el texto.

El documento se busca en el disco y van apareciendo las líneas de texto donde se encuen-

Banco de pruebas

tra el objeto de la búsqueda. Para mayor precisión, un cursor en forma de flecha señala la palabra o palabras que interesan. Vemos también en pantalla cómo se efectúa la sustitución.

Al final de todas estas operaciones, MICROSCRIPT pregunta si queremos que el documento sea actualizado en función de los cambios introducidos, por si solo se pretende practicar o averiguar como funciona esta opción.

Al efectuar muchos cambios en un documento, tal vez sea necesario recomponer el formato de acuerdo a las especificaciones originales del mismo, cómo se verá luego.

OPCION F (GESTION DE FICHERO)

Esta opción nos introduce en un submenú de 5 opciones, mediante las cuales podemos realizar:

1. Cambiar de nombre un fichero.
2. Suprimir un fichero del disco.
3. Copiar ficheros de un disco a otro.
4. Unir dos ficheros en uno.
5. Mostrar la lista de ficheros del disco.

Cada una de ellas posee sus propias instrucciones, por demás muy sencillas y que se reducen a dar un nombre de fichero o dos y pulsar ENTER.

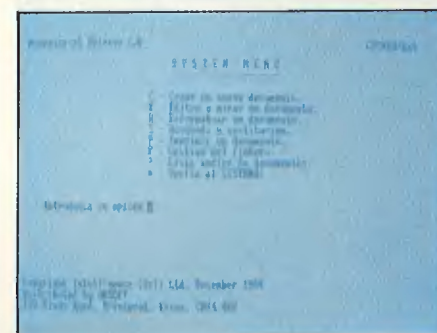
Hay que aclarar que, para copiar ficheros, se necesitan dos unidades de disco. No se puede pasar un fichero a otro disco distinto pero de la misma unidad.

Para realizar esta tarea, está el comando CP/M FILECOPY. COM en el disco con el sistema operativo que se entrega al adquirir el CPC664.

El resto de las opciones de este submenú se explican por sí mismas.

OPCION C (CREACION DE UN DOCUMENTO)

Al comenzar a trabajar con un nuevo documento, lógicamente tenemos que comunicarle al programa nuestra intención. Para ello,



TIME

Harum

Lorem ipsum dolor sit
elit, sed diam nonnum
labore et dolore ma
enim ad minimim ven

?

laborum e
er expedit
nobis elig
anim id qu
voluptas a
Tempore
tum rerum
renudian

au a
ego
eam
pau
ami
aug
odic
mod
epu
umc
coe
Invit
aeq
fact
con
velli

FDD

Si MICROSCRIPT está dividido en ficheros diferentes, que se cargan en memoria cuando se necesitan, muy bien podría incluirse otro con pantallas de ayuda que nos informarán de los comandos a nuestra disposición en todo momento, a la simple pulsación de una tecla.

Aunque esto tal vez ralentizará las operaciones del programa, habría que preguntarse si no merece la pena. Ustedes decidirán, según la acogida que dispensen a los programas de gestión que AMSTRAD ofrezca.

De la misma forma que se hizo con ANSWORD la semana pasada, vamos a intentar ofrecer una panorámica de lo que podemos realizar con MICROSCRIPT.

AMSWORD VS. MICROSCRIPT

	AMSWORD	MICROSCRIPT
SOPORTE	CINTA/DISCO	DISCO
EXTENSION	25 KBYTES	159 KBYTES
MEMORIA LIBRE PARA TEXTO	13 KBYTES	12 KBYTES (1 DISCO) 169 KBYTES (2 DISCOS)
MANUALES	ACEPTABLES (45 PAG.)	MENOS ACEPTABLES (87 PAG.)
USO	MUY SENCILLO	MAS COMPLEJO
POTENCIA	BUENA	REGULAR (1 DISCO) EXCELENTE (2 DISCOS)
VELOCIDAD DE ACCESO	INSTANTANEA	MUY BUENA
	TODO EL PROGRAMA ESTA EN MEMORIA	
DISCOS NECESARIOS	1 UNIDAD	1 UNIDAD PARA EXTRAER EL MAXIMO RENDIMIENTO, 2
PRECIO	CINTA: 2300 DISCO: 6500	DISCO: 12000

pulsando «C» en el menú principal de MICROSCRIPT, entramos en el proceso de crear un nuevo documento. Se nos pedirá el nombre del nuevo fichero en primer lugar (*no más de ocho caracteres más los tres del «apellido», CP/M estándar*) y, si no damos el identificador de unidad de disco, MICROSCRIPT supondrá que queremos crearlo en el disco por defecto del sistema en ese momento (*normalmente la unidad «A»*).

A continuación, se abrirá un fichero en el disco con el nombre que hayamos escogido y podemos comenzar a introducir texto en la pantalla.

Esta queda en blanco, sólo aparece un cursor para indicarnos la posición donde se verá el texto.

El editor de MICROSCRIPT es uno de los llamados «a pantalla completa», como debe ser el editor de cualquier procesador de textos que se precie, por lo que disponemos en él de las funciones típicas: movernos por toda la pantalla usando las teclas de movimiento del cursor, desplazarnos al comienzo o final de palabra, línea de texto o el propio fichero.

Esta opción nos atrevemos a decir que ha sido diseñada para meter texto a bulto, es decir, no se cuida en absoluto la apariencia, la estética, de lo que vemos en pantalla (*por ejemplo, no está justificado*).

De esa tarea se encarga otro subprograma de MICROSCRIPT, como luego veremos.

Disponemos, no obstante, de hasta 9 «reglillas» que marcan cosas como márgenes, topes de tabulación, posición de sangrado, etc.... Pueden utilizarse cuantas veces se deseen diferentes reglillas, que luego, al refor-

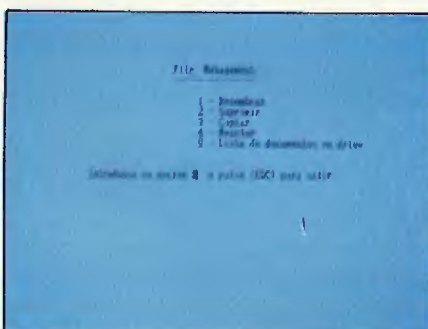
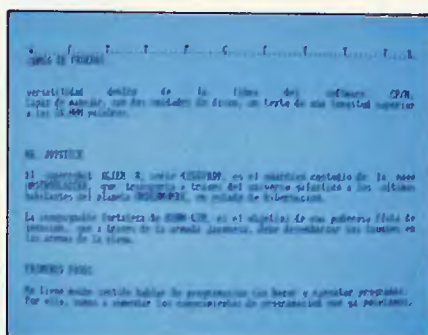
matar el documento, serán tenidas en cuenta para ajustar cada parte de texto a la apariencia deseada.

Por supuesto, nosotros podemos crear nuestras propias reglillas y almacenarlas en disco para su posterior empleo.

Con esta herramienta se pueden hacer verdaderas «composiciones» a la hora de asignar un formato a cualquier documento.

OPCION E (EDITAR UN DOCUMENTO)

Esta opción está pensada para, una vez escrito un documento, añadirle más texto y/o



modificarlo de acuerdo con nuestras necesidades.

Seguimos disponiendo de los mismos comandos que en el modo anterior, más algún otro que afecta a la corrección de errores.

OPCION R (REFORMATEAR UN DOCUMENTO)

Como su propio nombre indica, la opción R del menú principal nos permite dar forma a un documento una vez escrito y revisado. Diríamos que es el paso inmediatamente anterior a sacarlo por impresora.

Las posibilidades son innumerables, ya que prácticamente cada párrafo del texto puede tener su propio formato, dependiendo de la reglilla que contuviera o la que hagamos activa en ese momento.

Este es el momento de decidir los márgenes, el indentado o sangrado, si deseamos o no que el texto se justifique por la derecha o por la izquierda, etc.

Una vez terminado el proceso de dar forma al documento, ESC + E lo manda directamente al disco.

OPCION P (IMPRIMIR UN DOCUMENTO)

El cometido de esta opción resulta obvio. Se coloca aparte en el menú principal porque incluye más posibilidades que el mero hecho de sacar un fichero por la impresora, sin más.

El procedimiento es simple: tecleando el nombre del fichero, será traído del disco e impreso con los parámetros por defecto del programa.

Podemos fijar a nuestra conveniencia casi todos los parámetros de impresión: margen por la derecha y por la izquierda, subrayado y enguionado de texto, número de líneas por página, impresión o no de títulos, subtítulos, pies de página, número de las mismas, etc.

Para acceder a este submenú, basta pulsar ENTER, con lo que vemos en pantalla una lista con todos los parámetros de impresión. Si uno no nos convence, tecleamos el nuevo valor + ENTER. Para los que sí estén correctos, sólo ENTER.

También, antes de todo esto, podemos obtener un directorio de los ficheros del disco.

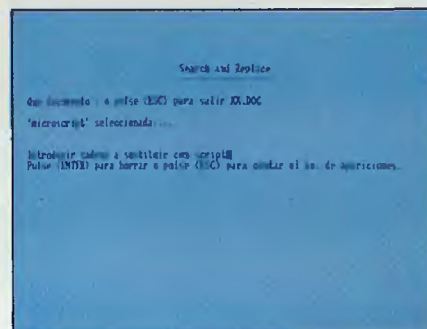
MICROSCRIPT es programable

Queda por comentar, por último dos interesantes características de MICROSCRIPT, que serán bienvenidas sobre todo por usuarios avanzados: la posibilidad de realizar cálculos numéricos y programas empleando los propios comandos de MICROSCRIPT, para automatizar trabajos complejos o que requieran ser llevados a cabo muchas veces.

La primera convierte a MICROSCRIPT es una primitiva hoja de cálculo: podemos encolumnar números y obtener totales de los mismos vertical y horizontalmente, además de imprimir o guardar en disco los resultados, por supuesto.

TABLA DE FUNCIONES DE MICROSCRIPT

COMANDO	FUNCION
ENTER	Retorno de carro.
DEL	El cursor retrocede un espacio y borra el carácter sobre el que se sitúa.
←	Cursor Izq.
→	Cursor Der.
↑	Cursor arriba.
↓	Cursor abajo.
TAB	Coloca el cursor en el siguiente tope de tabulación.
CTRL+N	Inserta texto.
CTRL+P	Desplaza el texto hacia la izquierda eliminando espacios en blanco.
CTRL+T	Avanza palabra.
CTRL+R	Retrocede palabra.
CTRL+Y	Borra línea.
ESC+O	Inserta líneas en blanco para poder introducir texto.
CTRL+→	Cursor a final de línea.
CTRL+←	Cursor al principio de línea.
ESC+↑	Cursor a la línea superior de pantalla.
ESC+↓	Cursor a la línea inferior de pantalla.
ESC+>	Pasa a mayúsculas todos los caracteres a la derecha del cursor.
ESC+<	A minúsculas todos a la izquierda del cursor.
ESC+{	Marca de inicio de bloque.
ESC+)+B	Marca de final de bloque. El texto se transfiere a una memoria intermedia.
ESC+)+L	Copia bloque.
ESC+)+R	Mueve bloque y destruye el anterior.
ESC+*+O	Copia bloque escribiendo encima del texto.
ESC+*+I	Inserta el bloque de texto contenido en la memoria temporal.
ESC+*+E	Lo mismo que el comando anterior, excepto que desplaza el texto existente para no escribir encima.
ESC+C	Centra texto.
ESC+J	Formatea el texto de acuerdo con la reglilla vigente.
ESC+S	Depositar reglilla en la memoria intermedia.
ESC+R	Recupera reglilla.
ESC+U	Indica a MICROSCRIPT que emplee una determinada reglilla.
ESC+—	Línea de guiones bajo la línea donde esté el cursor.
ESC+—	Subrayado.
CTRL+↓	Avance de página.
ESC+[Borra desde la posición del cursor hasta el final de la línea por la derecha.
ESC+]	Lo mismo, pero por la izquierda.
CTRL+V	Borra palabra.
CTRL+↓	Ir al comienzo del documento.
ESC+G	Mezclar el documento corriente con uno de disco.
ESC+I	Modo inserción.
ESC+ESC+I	Desplaza texto a la línea siguiente.
ESC+ESC+I	Abandonar el modo inserción.
ESC+ESC+I	Ir al final del texto.
ESC+×	Justificar una línea por la derecha.
ESC+F	Encontrar texto.
ESC+#	Guardar texto en una «memoria de abreviación».
ESC+@	Recupera texto de una de esas memorias.
ESC+M	Pasa a modo alternativo.
ESC+ESC+M	Abandona modo alternativo.
ESC+Q	Deja el documento que estaba siendo editado, SIN GUARDARLO EN DISCO.
ESC+Q	Carga documento.
ESC+E	Deja el documento GUARDANDOLO EN DISCO.
ESC+T	Ir al principio del documento.
COPY	Muestra en pantalla el fichero de información del sistema.
ESC+COPY	Copia una línea de texto en una memoria intermedia para poderla insertar luego en cualquier parte del documento.
ESC+B	Inserta dicha línea.
ESC+B	Cursor al final del presente documento.



La segunda es una clara prueba de la inmensa potencia que posee este programa. Cada comando de MICROSCRIPT puede ser representado de tal modo (*aparte del normal*) que pueden construirse programas muy complejos, con posibilidades interesantísimas, como por ejemplo, aceptar entrada desde teclado en medio de cualquier operación de disco y obrar en consecuencia según las nuevas instrucciones.

Todo este tema se encuentra detalladamente explicado, junto con ejemplos de uso, en el manual del usuario. Se le ha dedicado amplio espacio por la importancia que entrañan.

¡Ay!, los manuales

Respecto a los manuales que se entregan con el programa, la presentación es impecable. En cuanto al contenido, el panorama es algo más sombrío. No por falta de detalle en la enumeración de los comandos o en ejemplos de uso de los mismos, sino por una traducción del inglés como mínimo irregular, y en algunos pasajes, francamente deplorable.

Y la tabla comparativa...

Como colofón de este artículo, vamos a establecer una comparación entre los dos programas que hemos analizado: AMWORD y MICROSCRIPT, recordando siempre que cada uno de estos programas ha sido concebido para operar en una zona distinta del proceso de textos, respondiendo a necesidades diferentes.

FICHA DEL PROGRAMA MICROSCRIPT

CREADO POR:
INTELLIGENCE LTD. & AMSOFT
COMERCIALIZADO POR:
INDESCOMP
SOPORTE: **DISCO**
SISTEMA OPERATIVO: **CP/M**
PRECIO: **12.000**

Suscríbete... y uno de estos tres sensacionales juegos será tuyo... ¡GRATIS!

M.H. AMSTRAD te da a elegir entre tres de los mejores juegos existentes en el mercado para AMSTRAD; **COMBAT LYNX**, **DALEY THOMPSON'S DECATHLON** y **BEACH HEAD**, cualquiera de los cuales puede ser tuyo solamente con suscribirte a nuestra revista. **Aprovecha esta ocasión excepcional** y ahorra 2.100 pesetas (precio de venta del programa) más el importante descuento que se produce en el precio de cada número, por el hecho de ser suscriptor. Disfruta de las ventajas que supone recibir cómodamente tu revista a domicilio y de la seguridad de tener tu ejemplar aunque se haya agotado en los quioscos.

Envíanos tu boletín de suscripción y no le des más vueltas, el número de juegos para regalos de suscripción, aunque grande, es limitado, y estos se podrían agotar mientras lo estás pensando.

BEACH HEAD producido por U.S. GOLD es una misión de desembarco en una costa fuertemente defendida por las fuerzas aeronavales enemigas. Debes conducir tu flota hacia la bahía y repeler el ataque aéreo, si lo consigues tu siguiente obstáculo será una flotilla de destructores y acorazados, superada la cual desembarcarás tus anfibios en las arenas de la bahía, estos deben superar las defensas costeras y llegar a la fortaleza que es el objetivo final.

COMBAT LYNX simula una misión de defensa de unas bases atacadas por una división acorazada. Disponemos para enfrentarnos a ellos de un modernísimo helicóptero.

Este juego podría incluirse dentro del catálogo de los de estrategia, y su complejidad le dota de una gran dosis de adicción y belleza.

DALEY THOMPSON'S DECATHLON con este juego OCEAN enciende la llama olímpica y te reta a superar los récords de los campeonatos de todos los tiempos, el decathlon se desarrolla en dos días de competición y se compone de las siguientes pruebas:

PRIMER DIA: 100 m lisos, salto de longitud, lanzamiento de peso, salto de altura y 400 m lisos.

SEGUNDO DIA: 110 m vallas, lanzamiento de disco, salto con pértiga, lanzamiento de jabalina y los 1.500 m.

Nos vemos obligados a sustituir, dentro de los juegos que ofrecemos a nuestros suscriptores, el **POLE POSITION** por el **COMBAT LYNX**, debido a que la empresa distribuidora nos comunica la imposibilidad de obtenerlo en España a corto plazo.

Aquellas personas que hayan enviado la suscripción expresando su deseo de recibir el **POLE POSITION** recibirán, o habrán recibido ya, una carta en la que se facilita un cupón donde nos podrán hacer saber cuál de los 3 juegos prefieren. Muchas gracias.

Utiliza el cupón
adjunto a la revista
o suscríbete por
teléfono
(91) 733 50 12
(91) 733 50 16





HA LLEGADO **RPA Systems inc.**

**NUEVA SERIE II DE PROGRAMAS EN PASCAL
PARA AMSTRAD CPC 6128 Y PCW 8256**

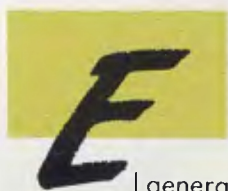
COMPATIBLE CPC 664

**BASE DE DATOS
HOJA DE CALCULO
CLIENTES - FACTURACION - MAILING
CONTABILIDAD GENERAL Y MUCHOS MAS.**

GENERADOR DE CARACTERES

Un programa como CHARGEN era necesario desde hace mucho tiempo para todos los usuarios de AMSTRAD.

CHARGEN se revela como una indispensable herramienta a la hora de personalizar nuestros propios programas, sean juegos, utilidades o lo que nuestra imaginación desee.



El generador de caracteres es un programa de utilidad, soberbiamente diseñado, que permite cualquier carácter del juego de caracteres del Amstrad a nuestro gusto. De hecho, podemos crear tantos juegos de caracteres como queramos.

Se incluyen gran cantidad de facilidades dentro del programa, como por ejemplo: rotación e inversión de los caracteres, efecto de espejo horizontal y vertical, etc.

Los datos numéricos de los nuevos caracteres pueden examinarse en pantalla o impresora y almacenarse en disco o cassette.

El programa muestra en pantalla una rejilla de 8x8, con un cursor parpadeante en la esquina superior izquierda.

El diseño de los caracteres se realiza rellenando las cuadrículas pertinentes pulsando el botón de fuego del joystick o la tecla COPY.

Obviamente, antes el cursor deberá moverse a la posición deseada mediante el joystick o las teclas de movimiento situadas alrededor de la tecla COPY.

Para borrar una de las celdillas, se sigue el mismo método que para iluminarlas.

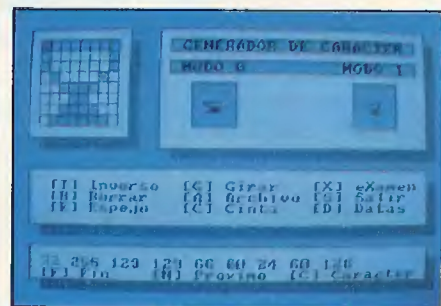
A medida que nuestro carácter se construye, a la derecha de la rejilla va apareciendo el diseño tal y como quedaría en los modos 0 y 1 respectivamente.

Al acabar la tarea, basta pulsar «A» para salvar el diseño en memoria; el nuevo carácter reemplazará el que el sistema posee por defecto con el mismo número. En este mo-

ESTRUCTURA DEL PROGRAMA

NUMERO DE LINEA FUNCION EN CHARGEN

10-25	Título.
30-50	Inutiliza la tecla ESC, selecciona el modo de pantalla, colores y asigna a las variables sus valores iniciales.
55-90.	Muestra el cursor en pantalla y espera a que se pulse una tecla.
95-150	Mueve el cursor en la dirección adecuada.
155-165	Estado del cursor (Activo/inactivo).
170-200	Opción girar.
205-225	Opción inverso.
230-285	Opción examen.
290-305	Opción borrar.
310-425	Salvar/cargar de cinta/disco.
430-480	Opción Archivo.
485-500	Opción salir.
505-550	Opción espejo.
555-690	Opción datos.
695-705	Introducir la matriz del carácter dentro de by%.
710-865	Carga del código máquina, definición de ventanas y gráficos.
870-875	Datos de código máquina para la rutina de transferencia de caracteres.



mento, pueden salvarse en cassette o disco los nuevos caracteres.

Todo el programa gira alrededor del concepto de menú, para facilitar al máximo su manejo.

Existe una ventana especialmente dedicada a mostrar todos los comandos del menú principal: para elegir cualquier opción basta con pulsar la letra entre corchetes correspondiente a la inicial del comando.

Cuando es necesario, en la llamada «área de información» aparece justamente eso, información adicional, que a veces se materializa en un segundo submenú de opciones, igualmente claro y sencillo de usar: basta apretar la tecla correspondiente a la opción deseada.

Antes comentamos que es posible salvar/cargar juegos de caracteres desde cinta/disco, y existe para ello la opción CINTA en el menú principal.

Dentro de ella, el programa pide en un momento dado pulsar PLAY y luego cualquier tecla. Los usuarios de disco no deben preocuparse de este mensaje: al pulsar una tecla la información es vertida automáticamente a disco.

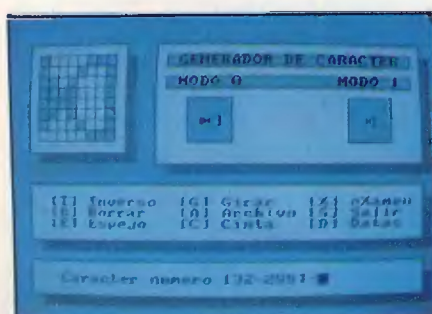
Lógicamente, si se tienen conectados simultáneamente disco y cassette, antes de ejecutar CHARGEN debe informarse al ordenador qué periférico va a usarse, mediante los comandos TAPE o DISC.

TABLA DE COMANDOS

NOMBRE	LO QUE HACE
INVERSO	Invierte los contenidos de los cuadros o celdas de la rejilla de diseño. Cualquier celda iluminada pasa a ser apagada y viceversa, es decir, produce un negativo de la imagen primitiva.
BORRAR	Limpia la rejilla. Si el símbolo que estamos manejando no ha sido guardado en la memoria mediante la opción Archivo, lo perderemos. Este comando debe usarse con cuidado.
ESPEJO	Produce la imagen especular del carácter que se encuentra en la rejilla respecto a los ejes horizontal o vertical.
GIRAR	Gira el carácter contenido en la rejilla 90 grados.
ARCHIVO	Almacena un carácter en memoria para posterior uso, o bien para ser volcado a cinta/disco.
CINTA	Graba en cinta/disco 2 FICHEROS, uno, los caracteres en formato binario y el segundo en formato ASCII conteniendo información necesaria para su posterior uso y manipulación por CHARGEN.
EXAMEN	Permite mostrar en pantalla cualquier carácter en el rango 32-255 para examinarlo o rediseñarlo.
SALIR	Fin del programa.
DATAS	Muestra los ocho bytes que forman un carácter en formato numérico decimal, para ser dirigidos a la [P]antalla o a la [I]mpresora. Este comando posee un submenú que permite examinar el siguiente carácter ([N]), uno cualquiera ([C]) y [F]inalizar.

VARIABLES PRINCIPALES

NOMBRE	SIGNIFICADO
gr%	Indica el estado de cada cuadrado de la rejilla empleada para diseñar los caracteres. Cuadrado encendido=1 Cuadrado apagado=0
ro%	Condición después de la rotación (1=activo, 0=activo).
re%	Chequeo para averiguar si un carácter está o no redefinido (1=si, 0=no).
by%	Los 8 bytes que forman la matriz de cada carácter.
gx%	Posición horizontal del cursor.
gy%	Posición vertical del cursor.
mx1%	Mode 0. Posición de comienzo del punto horizontal de «ploteado».
mx2%	Mode 1. Posición de comienzo horizontal del ploteado.
my1%	Mode 0. Comienzo del punto vertical de plot.
my2%	Mode 1. Comienzo del punto vertical de plot.
f	Flag para la pluma en la rejilla de diseño.
p	Flag para la pluma en la rejilla a tamaño real.



```

10 REM *****
15 REM GENERADOR DE CARACTERES
20 REM BY R.J.CROSKELL
25 REM *****
30 MODE 1:INK 0,10:INK 1,0:INK 2,20
:INK 3,15,1:PAPER 0:BORDER 10
35 CALL &BB48:IF PEEK(38500)=255 TH
EN 45 ELSE POKE 38500,255
40 SYMBOL AFTER 32:MEMORY 37999:SPE
ED INK 15,15
45 DIM gr%(8,8),ro%(8,8),re%(224),b
y%(8):gx%=1:gy%=1:mem%=&9470
50 f=0:p=0:mx1%=286:mx2%=534:my1%=2
78:my2%=278:KEY 140,"":GOSUB 710
55 PEN #5,3:LOCATE #5,gx%,gy%:PRINT
#5,CHR$(255);
60 k$=UPPER$(INKEY%):IF k$="" THEN
60
65 IF k$=CHR$(240) OR JOY(0)=1 THEN
95
70 IF k$=CHR$(241) OR JOY(0)=2 THEN
110
75 IF k$=CHR$(242) OR JOY(0)=4 THEN
125
80 IF k$=CHR$(243) OR JOY(0)=8 THEN
140
85 IF k$=CHR$(224) OR JOY(0)=16 THE
N 155
90 ON INSTR("GIXBCASED",k$) GOTO 17
0,205,230,290,310,430,485,505,555:G
OTO 60
95 IF gy%=1 THEN PRINT CHR$(7);:GOT
O 60 ELSE LOCATE #5,gx%,gy%
100 IF gr%(gx%,gy%)=1 THEN PEN #5,0
ELSE PEN #5,2
105 PRINT #5,CHR$(255);:gy%=gy%-1:G
OTO 55
110 IF gy%=8 THEN PRINT CHR$(7);:GO
TO 60 ELSE LOCATE #5,gx%,gy%
115 IF gr%(gx%,gy%)=1 THEN PEN #5,0
ELSE PEN #5,2
120 PRINT #5,CHR$(255);:gy%=gy%+1:G
OTO 55
125 IF gx%=1 THEN PRINT CHR$(7);:GO
TO 60 ELSE LOCATE #5,gx%,gy%
130 IF gr%(gx%,gy%)=1 THEN PEN #5,0
ELSE PEN #5,2
135 PRINT #5,CHR$(255);:gx%=gx%-1:G
OTO 55
140 IF gx%=8 THEN PRINT CHR$(7);:GO
TO 60 ELSE LOCATE #5,gx%,gy%
145 IF gr%(gx%,gy%)=1 THEN PEN #5,0
ELSE PEN #5,2
150 PRINT #5,CHR$(255);:gx%=gx%+1:G
OTO 55
155 IF gr%(gx%,gy%)=1 THEN gr%(gx%,
gy%)=0:p=0:ELSE gr%(gx%,gy%)=1:p=1
160 PLOT mx1%+(gx%*4)-2,my1%-(gy%*2
),p:PLOT mx1%+(gx%*4),my1%-(gy%*2)
165 PLOT mx2%+(gx%*2),my2%-(gy%*2):
GOTO 60
170 FOR t%=1 TO 8:FOR g%=8 TO 1 STE
P -1:c%=ABS(g%-9)
175 ro%(t%,c%)=gr%(g%,t%):NEXT:NEXT
180 FOR t%=1 TO 8:FOR g%=1 TO 8:gr%
(t%,g%)=ro%(t%,g%)
185 IF gr%(t%,g%)=1 THEN PEN #5,0:p
=1 ELSE PEN #5,2:p=0
190 LOCATE #5,t%,g%:PRINT #5,CHR$(2
55);:PLOT mx1%+(t%*4)-2,my1%-(g%*2
),p
195 PLOT mx1%+(t%*4),my1%-(g%*2):PL
OT mx2%+(t%*2),my2%-(g%*2):NEXT:NEX
T
200 gx%=1:gy%=1:GOTO 55
205 FOR t%=1 TO 8:FOR g%=1 TO 8:LOC
ATE #5,t%,g%
210 IF gr%(t%,g%)=1 THEN gr%(t%,g%)
=0:f=2:p=0:ELSE gr%(t%,g%)=1:f=0:p=
1
215 PEN #5,f:PRINT #5,CHR$(255);:PL
OT mx1%+(t%*4)-2,my1%-(g%*2),p
220 PLOT mx1%+(t%*4),my1%-(g%*2):PL
OT mx2%+(t%*2),my2%-(g%*2)
225 NEXT:NEXT:gx%=1:gy%=1:GOTO 55
230 CLS #2:LOCATE #2,2:INPUT #2,"
Caracter a cargar (32-255)-",char$
235 IF VAL(char$)<32 OR VAL(char$)>
255 THEN 230 ELSE CLS #2
240 v%=VAL(char$):v%=(v%-32)*8:FOR

```



```

t%=0 TO 7:by%(t%+1)=PEEK(39000+u%+t%):NEXT
245 FOR t%=1 TO 8:r%=1:FOR g%=7 TO 0 STEP -1
250 IF by%(t%)=2*g% THEN 260
255 gr%(r%,t%)=0:GOTO 265
260 by%(t%)=by%(t%)-2*g%:gr%(r%,t%)=1
265 r%=r%+1:NEXT:NEXT:FOR t%=1 TO 8
:FOR g%=1 TO 0
270 IF gr%(t%,g%)=1 THEN PEN #5,0:p=1:ELSE PEN #5,2:p=0
275 LOCATE #5,t%,g%:PRINT #5,CHR$(255);:PLOT mx1%+(t%*4)-2,my1%-(g%*2),p
280 PLOT mx1%+(t%*4),my1%-(g%*2):PLOT mx2%+(t%*2),my2%-(g%*2):NEXT:NEXT
285 qx%=1:gy%=1:LOCATE #2,8,2:PRINT #2,"** AREA DE INFORMACION **":GOTO 0 55
290 LOCATE #5,1,1:PEN #5,2:PRINT #5,STRING$(64,255);
295 FOR t%=6 TO 9:LOCATE #4,4,t%:PRINT #4,STRING$(4,32);
300 LOCATE #4,19,t%:PRINT #4,STRING$(4,32);:NEXT
305 ERASE gr%:DIM gr%(8,8):gx%=1:gy%=1:GOTO 55
310 CLS #2:PEN #2,1:LOCATE #2,4,1:PEN #2,"Antes de realizar una grabacion"
315 LOCATE #2,4,2:PRINT #2,"archivos los caracteres con [S] ?"
320 LOCATE #2,12,3:PRINT #2,"Si o No (S or N).":
325 k%=UPPER$(INKEY$):IF k%="S" OR k%="N" THEN 330 ELSE 325
330 ON INSTR("SN",k%) GOTO 335,425
335 CLS #2:LOCATE #2,4,2:PRINT #2,"Salvar o Cargar Caracteres (S o C)?"
340 k%=UPPER$(INKEY$):IF k%="S" OR k%="C" THEN 345 ELSE 340
345 IF k%="S" THEN 390
350 CLS #2:LOCATE #2,2,2:INPUT #2,"Nombre [8 Max]-",name$
355 IF LEN(name$)>8 THEN 350 ELSE CLS #2:name$=UPPER$(name$)
360 LOCATE #2,6,2:PRINT #2,"Pulsa P LAY, y luego una tecla":WHILE INKEY$="":WEND
365 CLS #2:LOCATE #2,7,2:PRINT #2,"Cargando [":name$;"]:"
370 LOAD"!"+name$+".OPENIN"!"+name$+ ".CHR":FOR t%=1 TO 224:INPUT #9,re%(t%):NEXT:CLOSEIN
375 CLS #2:LOCATE #2,9,2:PRINT #2,"Programa en memoria."
380 FOR t=1 TO 3000:NEXT:CLS #2:LOCATE #2,8,2
385 PRINT #2,"** AREA DE INFORMACION **":GOTO 60
390 CLS #2:LOCATE #2,2,2:INPUT #2,"Nombre [8 Max]-",name$
395 IF LEN(name$)>8 THEN 390 ELSE CLS #2:name$=UPPER$(name$)
400 CLS #2:LOCATE #2,6,2:PRINT #2,"Pulsa REC+PLAY y una tecla"
405 WHILE INKEY$="":WEND:CLS #2:LOCATE #2,7,2:PRINT #2," Salvando [":name$;"]:"
410 SAVE"!"+name$+".b,39000,1792:OPEN OUT"!"+name$+ ".CHR":FOR t%=1 TO 224:WRITE #9,re%(t%):NEXT
415 CLOSEOUT:CLS #2:LOCATE #2,12,2:PRINT #2,"Caracteres Salvados"
420 FOR t=1 TO 3000:NEXT
425 CLS #2:LOCATE #2,8,2:PRINT #2,"** AREA DE INFORMACION **":GOTO 60
430 CLS #2:LOCATE #2,4,2:INPUT #2,"Caracter numero [32-255]-",char$
435 IF VAL(char$)<32 OR VAL(char$)>255 THEN 430 ELSE CLS #2
440 v%=VAL(char$):IF re%(v%-31)=0 THEN 465 ELSE CLS #2
445 LOCATE #2,3,2:PRINT #2,"Caracter ya definido, cambio [S/N]"
450 k%=UPPER$(INKEY$):IF k%="S" OR k%="N" THEN 455 ELSE 450
455 IF k%="S" THEN 465 ELSE CLS #2:LOCATE #2,8,2
460 PRINT #2,"** AREA DE INFORMACION **":GOTO 60
465 GOSUB 695:v%=(v%-32)*8:FOR t%=0 TO 7

```

```

470 POKE 39000+u%+t%,by%(t%+1):NEXT
CLS#2:LOCATE #2,11,2
475 PRINT #2,"Caracter Archivado":re%(u%/8+1)=1:FOR t=1 TO 3000
480 NEXT:CLS #2:LOCATE #2,8,2:PRINT #2,"** AREA DE INFORMACION **":GOTO 60
485 CLS #2:PEN #2,1:LOCATE #2,7,2:PRINT #2,"Salir del generador [S/N]"
490 k%=UPPER$(INKEY$):IF k%="S" OR k%="N" THEN 495 ELSE 490
495 IF k%="S" THEN MODE 1:PRINT"Programa desconectado":END
500 CLS #2:LOCATE #2,8,2:PRINT #2,"** AREA DE INFORMACION **":GOTO 60
505 CLS #2:LOCATE #2,3,2:PRINT #2,"Plano Vertical o Horizontal (V/H)"
510 k%=UPPER$(INKEY$):IF k%="V" OR k%="H" THEN 515 ELSE 510
515 IF k%="V" THEN 525 ELSE FOR t%=8 TO 1 STEP -1:FOR g%=1 TO 8
520 re%(g%,t%)=gr%(g%,ABS(t%-9)):NEXT:NEXT:GOTO 530
525 FOR t%=8 TO 1 STEP -1:FOR g%=1 TO 8:re%(t%,g%)=gr%(ABS(t%-9),g%):NEXT:NEXT
530 FOR t%=1 TO 8:FOR g%=1 TO 8:gr%(t%,g%)=re%(t%,g%):LOCATE #5,t%,g%
535 IF gr%(t%,g%)=1 THEN PEN #5,0:p=1 ELSE PEN #5,2:p=0
540 PRINT #5,CHR$(255);:PLOT mx1%+(t%*4)-2,my1%-(g%*2),p
545 PLOT mx1%+(t%*4),my1%-(g%*2):PLOT mx2%+(t%*2),my2%-(g%*2):NEXT:NEXT
550 CLS #2:LOCATE #2,9,2:PRINT #2,"** AREA DE INFORMACION **":GOTO 55
555 code%=32:CLS #2:LOCATE #2,6,2:PRINT #2,"Por Pantalla o Impresora (P/I)?"
560 k%=UPPER$(INKEY$):IF k%="P" OR k%="I" THEN 565 ELSE 560
565 IF k%="I" THEN 615 ELSE CLS #2:FOR t%=0 TO 223:IF re%(t%+1)=0 THEN 570 ELSE
570 CLS #2:FOR t%=(code%-32) TO 223:IF re%(t%+1)=0 THEN 585
575 CLS #2:LOCATE #2,1,2:PEN #2,0:PEN #2,2,STR$(t%+32);:PEN #2,1:FOR g%=0 TO 7
580 a%=(t%*8):p%=PEEK(39000+a%+g%):PRINT #2,STR$(p%);:NEXT:GOTO 590
585 CLS #2:LOCATE #2,13,2:PEN #2,1:PRINT #2,STR$(t%+32);:V A C I O
590 LOCATE #2,2,3:PEN #2,1:PRINT #2,"[F] Fin [N] Proximo [C] Caracter"
595 k%=UPPER$(INKEY$):IF k%="F" OR k%="N" OR k%="C" THEN 600 ELSE 595
600 ON INSTR("FNC",k%) GOTO 675,605,610
605 NEXT:GOTO 675
610 GOSUB 680:GOTO 570
615 CLS #2:PEN #2,1:LOCATE #2,1,2:PEN #2,2,"Prepara la impresora y pulsa una tecla"
620 WHILE INKEY$="":WEND:FOR t%=0 TO 223:IF re%(t%+1)=0 THEN 580
625 FOR t%=(code%-32) TO 223:IF re%(t%+1)=0 THEN 645
630 CLS #2:LOCATE #2,1,2:PEN #2,0:PEN #2,2,STR$(t%+32);:PEN #2,1:PRINT #2,"SYMBOL";STR$(t%+32);:FOR g%=0 TO 7:a%=(t%*8):p%=PEEK(39000+a%+g%)
640 PRINT #2,STR$(p%);:PRINT #8,STR$(p%);:NEXT:PRINT #8,CHR$(13):GOTO 650
645 CLS #2:LOCATE #2,13,2:PEN #2,1:PRINT #2,STR$(t%+32);:V A C I O
650 LOCATE #2,2,3:PEN #2,1:PRINT #2,"[F] Fin [N] Proximo [C] Caracter"
655 k%=UPPER$(INKEY$):IF k%="F" OR k%="N" OR k%="C" THEN 660 ELSE 655
660 ON INSTR("FNC",k%) GOTO 675,665,670
665 NEXT:GOTO 675
670 GOSUB 680:GOTO 625
675 CLS #2:LOCATE #2,7,2:PRINT #2,"** AREA DE INFORMACION **":GOTO 55
680 LOCATE #2,1,2:PRINT #2,CHR$(18);:LOCATE #2,2,2:PEN #2,1
685 INPUT #2,"Caracter numero (32 to 255) ",code%

```

```

690 IF code%<32 OR code%>255 THEN 680 ELSE RETURN
695 FOR t%=1 TO 8:q%=0:z%=0:FOR g%=8 TO 1 STEP -1
700 IF gr%(g%,t%)<>1 THEN 705 ELSE z%=z%+2*q%
705 q%=q%+1:NEXT:by%(t%)=z%:NEXT:RETURN
710 RESTORE 870:FOR t%=0 TO 18:READ d:POKE mem%+t%,d:NEXT:CALL mem%
715 SYMBOL 255,254,254,254,254,254,254,254,0
720 WINDOW #1,2,39,15,19:PAPER #1,2:CLS #1
725 WINDOW #2,2,39,22,24:PAPER #2,2:CLS #2
730 WINDOW #3,2,11,2,11:PAPER #3,2:CLS #3
735 WINDOW #4,15,39,2,12:PAPER #4,2:CLS #4
740 WINDOW #5,3,10,3,10:PAPER #5,2:CLS #5
745 PEN 1:LOCATE 3,20:PRINT STRING$(38,207):LOCATE 3,25:PRINT STRING$(38,207);
750 LOCATE 3,12:PRINT STRING$(10,207):LOCATE 16,13:PRINT STRING$(25,207)
755 FOR t%=16 TO 19:LOCATE 40,t%:PRINT CHR$(207):NEXT
760 FOR t%=3 TO 11:LOCATE 12,t%:PRINT CHR$(207):NEXT
765 FOR t%=3 TO 12:LOCATE 40,t%:PRINT CHR$(207):NEXT
770 FOR t%=23 TO 24:LOCATE 40,t%:PRINT CHR$(207):NEXT
775 FOR t%=31 TO 159 STEP 16:PLOT t%,368,1:DRAW 0,-128:NEXT
780 FOR t%=368 TO 240 STEP -16:PLOT 31,t%:DRAW 128,0:NEXT
785 PLOT 15,15:DRAW 610,0:DRAW 0,50:DRAW -610,0:DRAW 0,-50
790 PLOT 15,384:DRAW 162,0:DRAW 0,-162:DRAW -162,0:DRAW 0,162
795 PLOT 15,94:DRAW 610,0:DRAW 0,82:DRAW -610,0:DRAW 0,-82
800 PLOT 624,384:DRAW -402,0:DRAW 0,-178:DRAW 402,0:DRAW 0,178
805 PLOT 238,368:DRAW 370,0:DRAW 0,-18:DRAW -370,0:DRAW 0,18
810 PLOT 238,336:DRAW 370,0:DRAW 0,-18:DRAW -370,0:DRAW 0,18
815 PLOT 270,304:DRAW 66,0:DRAW 0,-66:DRAW -66,0:DRAW 0,66
820 PLOT 510,304:DRAW 66,0:DRAW 0,-66:DRAW -66,0:DRAW 0,66
825 LOCATE #5,1,1:PRINT #5,CHR$(22)+CHR$(1)
830 PEN #4,1:PAPER #4,0:LOCATE #4,2,2:PRINT #4,"GENERADOR DE CARACTER"
835 LOCATE #4,2,4:PRINT #4," MODO 0
MODO 1 "
840 FOR t%=6 TO 9:LOCATE #4,4,t%:PRINT #4," "
845 LOCATE #4,19,t%:PRINT #4," "
:NEXT
850 PEN #2,1:LOCATE #2,8,2:PRINT #2,"** AREA DE INFORMACION **"
855 PEN #1,1:LOCATE #1,3,2:PRINT #1,"[I] Inverso [G] Girar [X] eXamen"
860 LOCATE #1,3,3:PRINT #1,"[B] Borrar [A] Archivo [S] Salir"
865 LOCATE #1,3,4:PRINT #1,"[E] Espajo [C] Cinta [D] Datos":RETURN
870 DATA &01,&00,&07,&21,&00,&A5,&11,&58,&98,&7E
875 DATA &12,&08,&23,&13,&78,&B1,&20,&F7,&C9

```



P ara que tus dedos no realicen el trabajo duro, M.H. AMS. TRAD lo hace por ti. Todos los listados que incluyen este logotipo se encuentran a tu disposición en un cassette mensual, solicitálos.

COMPUTER CENTER

COMANDANTE ZORITA, 13
28003 MADRID

TELS.: (91) 233 07 35
(91) 233 07 81

AMSTRAD 464 Verde	57.900 ptas.
AMSTRAD 664 Verde	99.900 ptas.
DISKETTE 3"	895 ptas.
INTERFACE DISCO 5.25"	5.900 ptas.
CINTA C-15 ESPECIAL ORDEN	85 ptas.

SOFTWARE ENTRETENIMIENTO (CASSETTE)

COMBAT LYNX	1.925 ptas.
DEAT PIT	1.925 ptas.
ALIEN-8	1.875 ptas.
KNIGHT LORE	1.875 ptas.
HARRIER ATTACK	995 ptas.
JUMP JET	2.695 ptas.
SIMULAD. VUELO 737	1.795 ptas.
FRUIT MACHINE	995 ptas.

SOFTWARE GESTION (DISCO)

CONTAB. GRAL.	11.995 ptas.
CONTROL STOCK	6.995 ptas.
BASE DE DATOS	6.995 ptas.

LIBROS

CURSO AUTODIDACTICO BASIC AMSTRAD (Contiene manual y dos cassettes)	2.695 ptas.
HACIA LA INTELIGENCIA ARTIF.	1.300 ptas.
40 JUEGOS EDUCATIVOS	800 ptas.
MUSICA Y SONIDO PARA AMSTRAD	1.300 ptas.
PROGRAMANDO CON AMSTRAD	1.900 ptas.

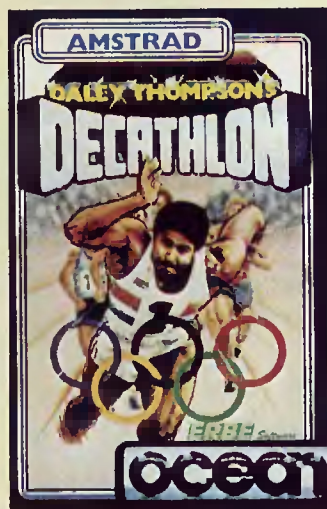
Tu pedido lo puedes recibir contra-reembolso (libre de gastos), llamando a los teléfonos (91) 233 07 35 y (91) 233 07 81.



1.975 Ptas.



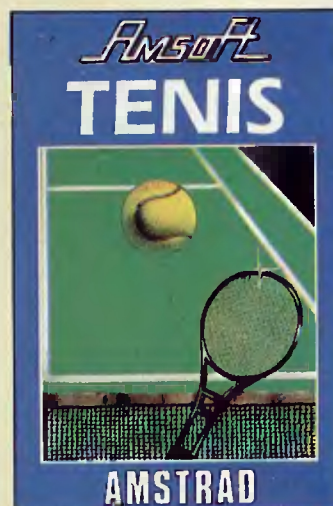
1.495 Ptas.



1.495 Ptas.



1.975 Ptas.



1.975 Ptas.



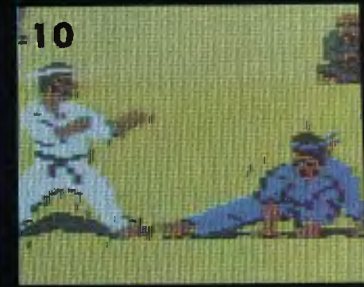
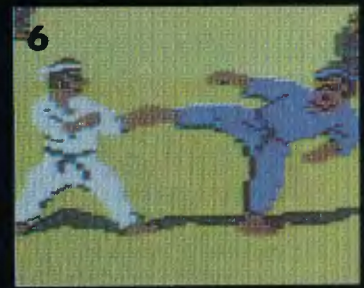
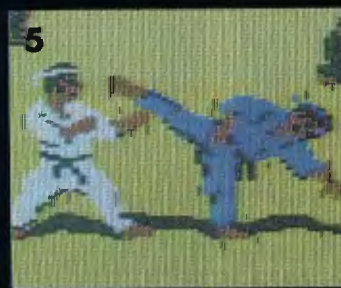
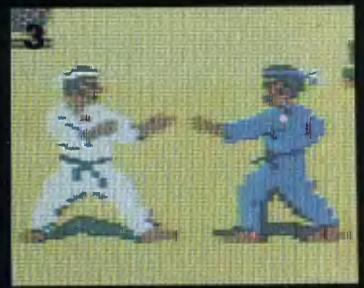
1.495 Ptas.

exploding fist





Mr. Joystick



exploding fist

Dentro del complejo e intrincado mundo de las artes marciales, el kárate es la disciplina más popular y difundida por los distintos países.

E

l kárate es un antiguo arte que nació en OKINAWA, durante una invasión de los ejércitos de uno de los emperadores de la enigmática China.

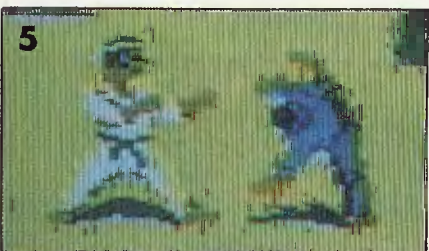
Como pueblo invasor los chinos prohibieron la tenencia y uso de armas a los habitantes del país, de modo que el dominio de la fuerza era fácil de mantener, sobre unos súbditos desarmados y sometidos.

Ante esta situación, los oprimidos bajo el yugo de los hijos del celeste imperio, se vieron obligados a desarrollar un sistema de lucha utilizando brazos y piernas, que les permitiese defenderse y atacar por sorpresa a hombres armados, constituyéndose de esta forma el origen de lo que ahora conocemos como KARA-TE.

Con la llegada a Europa de los maestros japoneses y koreanos, y su rápida difusión, se ha convertido en un deporte de competición, perdiendo totalmente su carácter de arte y pasando a tener como principal motivación la lucha por una medalla.

MELBOURNE HOUSE, siguiendo la clara tendencia deportiva de muchas firmas de software, ha creado una magnífica versión de kárate para ordenador personal. Tarea bastante difícil, debido a que la belleza de esta sublime disciplina reside en la variedad de golpes y posiciones que los luchadores pueden adoptar durante un combate.

El juego tiene como base una pelea entre dos luchadores. Permitiendo las distintas opciones de jugar contra el ordenador, o jugar dos personas entre sí.



Nuestro rival ha de ser vencido a toda costa y para ello podemos utilizar técnicas de ataque o de contragolpe, como si de un combate real se tratara.

Debemos elogiar ampliamente el trabajo de MELBOURNE HOUSE, porque su equipo de programadores y diseñadores de gráficos han conseguido en esta ocasión un producto auténticamente excepcional.

Lo verdaderamente asombroso del **EXPLODING FIST**, es el movimiento de los karatecas. Cualquier aficionado o asiduo practicante de las artes marciales, observará que las técnicas utilizadas por los luchadores son perfectas, la ejecución de cada golpe es un reflejo auténtico de la realidad y el dinamismo y ritmo del combate con ataques, sucesivas defensas e intercambio de golpes no puede ser más acertado.

No sería ninguna temeridad asegurar que el desarrollo del programa, haya estado asesorado por un maestro del kárate, por que de lo contrario sería difícil imaginar cómo se ha conseguido la fiel reproducción de golpes y técnicas.

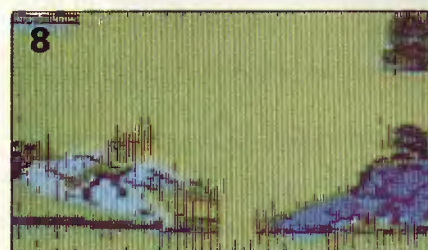
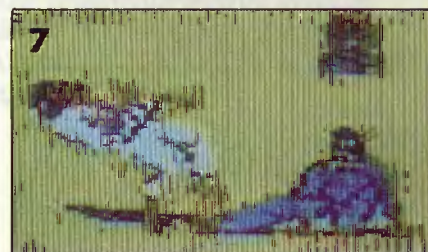
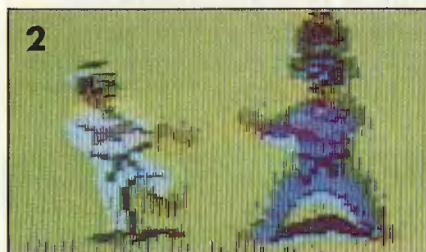
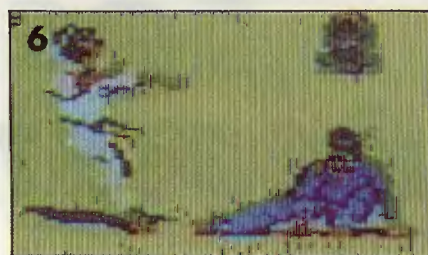
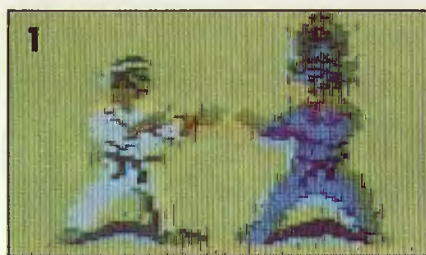
Para colmo, el jugador dispone de 18 golpes y técnicas distintas que puede utilizar en la lucha.

Todas ellas estudiadas minuciosamente y con una animación perfecta.

Si elegimos la lucha contra el ordenador, cada nuevo adversario lucha con mayor pericia, haciendo más difícil su derrota.

Los niveles que contempla el juego son desde principiante hasta décimo dan. Un combate se gana cuando se consiguen dos ipones, cada golpe ejecutado con acierto sobre nuestro adversario, incrementa el número de puntos, una forma adictiva para conseguir nuestro récord; bien sea por los puntos alcanzados, o por el dan conseguido.

Felicitaciones a MELBOURNE HOUSE.



TECNICAS DE COMBATE

Golpe de puño a la cara.
Golpe de puño al plexo solar.
Golpe de puño al estómago.
Patada frontal a la cara.
Patada circular a la cara.
Patada lateral a la cara.
Patada volante a la cara.
Patada lateral al estómago.
Patada baja a la rodilla.
Bloqueo de golpes con el antebrazo.
Barrido circular.
Defensa en salto contra barridos.
Avance hacia el enemigo.
Defensa retrocediendo.
Giro para cambiar de posición.

GYAKU TSUKI SHUDAN
OI TSUKI SHUDAN
OI TSUKI GEDAN
YOKO GERI SHUDAN
URA MAWASI GERI SHUDAN
MAWASHI GERI SHUDAN
YOKO TOBI GERI SHUDAN
MAWASHI GERI GEDAN
KIN GERI
UCHI UKE

Los REGISTROS (II)

El Z80, microprocesador que rige el funcionamiento del Amstrad, posee en su interior una posición de memoria, llamado el REGISTRO ACUMULADOR, el cual puede almacenar en su interior un solo byte de datos.

Y

a sabemos cómo cargar el acumulador con un número comprendido entre 0 y 255, mediante la instrucción:

LD A,N

cuyo código de operación es 3E hexadecimal.

Echemos una mirada al siguiente programa:

DIRECCION	CODIGO DE OPERACION	MNEMONICOS
3000	3E 07	LD A,7
3002	CD 5A BB	Cal &BB5A
3005	C9	RET

Los que se hayan tomado la molestia de teclear el HEXER, podrán introducir los códigos de operación mediante la opción 2 del mismo. Los que no lo tengan todavía listo para funcionar, pueden introducir el código mediante la sentencia POKE a partir de la dirección &3000 con:

POKE &3000,&3E

POKE &3001,&7

etc.

Una vez ejecutado este programa, escucharemos un sonido. Esto se debe a que hemos cargado el acumulador con el número 7 (código ASCII para el «beep») y, acto seguido, llamamos a la rutina TXT_OUTPUT del firmware (dirección &BB5A).

Esta rutina escribe en pantalla el carácter contenido en el acumulador, o, si el carácter no es imprimible porque se trata de un código de control (como 7), lo obedece.

De alguna forma, TXT_OUTPUT es el equivalente en código máquina de la sentencia Basic PRINT.

Es relativamente fácil imitar, desde lenguaje máquina, sentencias Basic, con la gran ventaja que trabajarán con mucha mayor rapidez. Por ejemplo, analicemos este pequeño programa que imita la sentencia INPUT:

DIRECCION	CODIGO DE OPERACION	MNEMONICOS
3000	CD 18 BB	CALL &BB1B
3003	CD 5A BB	CAL &BB5A
3006	C9	RET

Como se puede observar, recurrimos a dos rutinas del firmware del Amstrad.

La primera, localizada en la dirección &BB1B (no confundir el 8 último con la B) espera hasta que una tecla se pulse, y deja su código ASCII en el acumulador.

La segunda, nuestra vieja conocida TXT_OUT, toma ese valor del acumulador y lo escribe en la pantalla.

En el futuro veremos rutinas un poco más complejas, capaces de imitar e incluso de superar con gran ventaja, a sentencias del Basic Amstrad. De momento, podemos experimentar con esta variando los valores del acumulador; ¡a ver que pasa!

Los mnemónicos simplifican la programación

Ya hemos visto que, desde el momento que los códigos de operación son difíciles de recordar para las personas, usamos los mnemónicos para simplificar las cosas. Por ejemplo:

CALL &BB1B

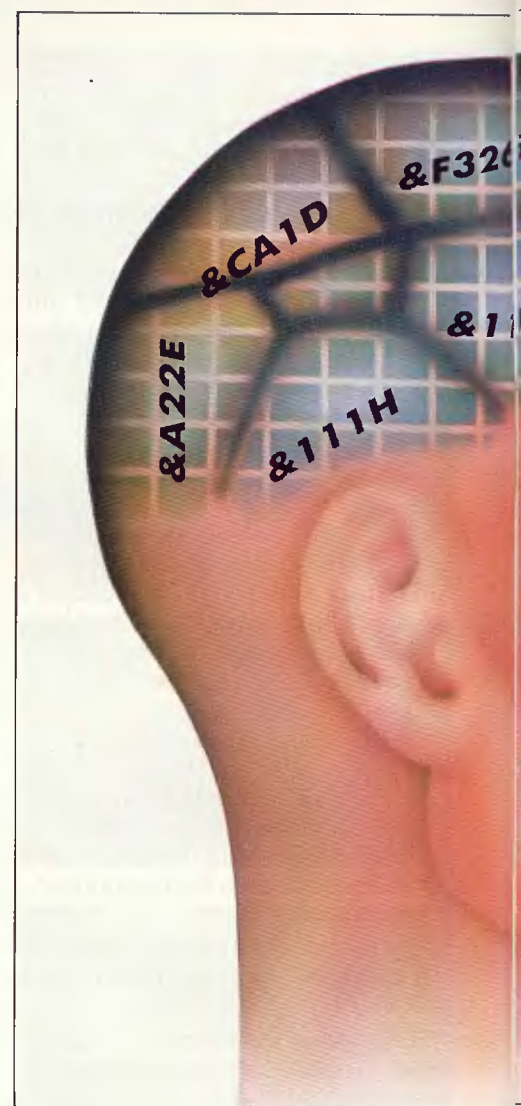
es mucho más legible que:

CD 18 BB

Todavía es factible dar un paso más para aumentar la legibilidad de nuestros programas, si ponemos nombres (ETIQUETAS) a ciertas partes críticas del mismo.

Así, a las rutinas que comienzan en &BB1B y &BB5A podríamos llamarlas «ESPERAR» y «PRINT», por ejemplo, con lo que tendríamos:

DIRECCION	CODIGO DE OPERACION	MNEMONICOS
3000	CD 18 BB	CALL ESPERAR
3003	CD 5A BB	CALL PRINT
3006	C9	RET



El programa queda ahora mucho más legible que antes, pues, aunque al ordenador le da lo mismo, para las personas está mucho más claro CALL PRINT que CALL &BB5A.

Respecto a las ETIQUETAS, hay que puntualizar unas cuantas cosas:

1. Las ETIQUETAS sólo pueden emplearse con los programas ensambladores, aunque nosotros las utilizaremos a veces, dando por sentado que todo el mundo sabe el número hexadecimal que representan.

2. Los códigos de operación no cambian a pesar de incluir la etiqueta, porque se supone que el valor que representa dicha etiqueta (PRINT equivale a &BB5)

Manipulando el acumulador

Una vez «metido» un byte en el acumulador, podemos hacer con él un montón de cosas útiles, como comentábamos en artículos anteriores.



Por ejemplo:

DIRECCION	CODIGO DE OPERACION	MNEMONICO
3000	CD 18 BB	CALL ESPERAR
3003	3C	INC A
3004	CD 5A BB	CALL PRINT
3007	C9	RET

Este programa incluye una nueva instrucción, cuyo mnemónico es INC a (código de operación &3C), que toma el número contenido en el acumulador e INCREMENTA su valor en una unidad.

Es decir, el programa espera hasta que pulsemos una tecla, incrementa el acumulador e imprime en pantalla el carácter correspondiente. Si pulsamos la tecla «A»(ASCII 65), aparecerá en pantalla «B»(ASCII 66).

Os sugerimos, como ejercicio, intentar que, al pulsar la «A», se vean en pantalla la «B» y la «C». Es fácil; basta INCREMENTAR el acumulador dos veces en el lugar adecuado.

No causará mucha sorpresa el descubrir la existencia de la instruc-

ción opuesta, para DECREMENTAR el acumulador.

Su mnemónico es DEC A, y su código de operación &3D.

Después de esto, lo que hace el siguiente programa resulta claro:

DIRECCION	CODIGO DE OPERACION	MNEMONICOS
300	CD 18 BB	CALL ESPERAR
3003	3D	DEC A
3004	CD 5A BB	CALL PRINT
3007	C9	RET

Otro buen ejercicio, para practicar y comprender a fondo el uso de estas dos instrucciones, sería ver en pantalla «ABC» al pulsar «B».

Todo lo que hay que hacer es llamar a la rutina ESPERAR, DECREMENTAR el acumulador, llamar a PRINT, INCREMENTAR el acumulador, etc.

Aunque el Z80 posee más de un registro, ya se comentó que el acumulador es, tal vez con excepción de otro que estudiaremos posteriormente, el más versátil. Vamos a aprovecharnos de esta facilidad para realizar PEEK y POKE desde código máquina.

Para el comando POKE, en Basic, era necesario especificar dos cosas:

- Dirección de memoria que va a ser alterada.
- El número que interesa colocar en ella.

De esta forma introducimos en memoria nuestros primeros programas en máquina (POKE &3000,&3E, por ejemplo).

El equivalente en código máquina es:

LD (&2FF9), A

es decir, se coge el número almacenado en el acumulador y se le transfiere a la posición de memoria &2FF9.

El mnemónico es LD (NN), A y el código de operación:

32 N N

en donde NN es la dirección que estamos «pokeando» en el orden habitual. Concretamente:

32 F9 2F

para la dirección &2FF9.

Como en el caso de POKE, especificamos la dirección a alterar y el número que la altera, dentro del acumulador.

Por tanto:

LD (&2FFA), A

debe leerse CARGA EL CONTENIDO DE LA DIRECCION DE MEMORIA &2FFA CON EL CONTENIDO DEL ACUMULADOR.

Código máquina

El siguiente programa muestra cómo puede utilizarse esta idea:

DIRECCION	CODIGO DE OPERACION	MNEMONICO
3000	3E 20	LD A,&20
3002	32 F9 2F	LD (&2FF9),A
3005	C9	RET

Si ejecutamos el programa y examinamos la posición de memoria &2FF9 mediante el Hexer, su contenido será &20.

Análogamente, el equivalente de PEEK también existe:

LD A,(&2FF8)

Cuando esta instrucción se haya ejecutado, el acumulador almacenará el número contenido en &2FF8. Su código de operación es &3A, y su mnemónico LD A,(NN).

Obsérvense los paréntesis en el mnemónico; son muy importantes. Mediante ellos diferenciamos entre DIRECCIONES DE MEMORIA Y CONTENIDO DE DIRECCIONES DE MEMORIA (los paréntesis se refieren a esta última

Transferencia de datos entre memorias

Podemos emplear este juego de instrucciones para combinar las funciones PEEK y POKE, es decir, somos capaces de TRANSFERIR DATOS ENTRE MEMORIAS a través del registro acumulador, como en este programa ejemplo:

DIRECCION	CODIGO DE OPERACION	MNEMONICOS
3000	3A F8 2F	LD A,(&2FF8)
3003	32 F9 2F	LD (&2FF9),A
3006	C9	RET

Mediante dos pasos, hemos copiado el contenido de la posición de memoria &2FF8 en la &2FF9 vía registro A.

Para comprobarlo, se pueden introducir mediante el Hexer dos números DISTINTOS en las posiciones de memoria &2FF8 y &2FF9.

A continuación, tras ejecutar el programa, examinando el contenido de &2FF9 veremos que es el mismo que el &2FF8 (*por favor, compruébese*).

Aprovechamos la coyuntura para recordar que Hexer toma por defecto la dirección &2FF8 para examinar código en lugar de &3000, que es donde comienza el programa por defecto. La razón de ello es usar las posiciones de memoria anteriores para poder ver los resultados de un programa, justo como acabamos de hacer.

Según la terminología «oficial», reservamos bytes para que el programa los use como ESPACIO DE TRABAJO (del inglés WORKSPACE, como aparece en más de un libro).

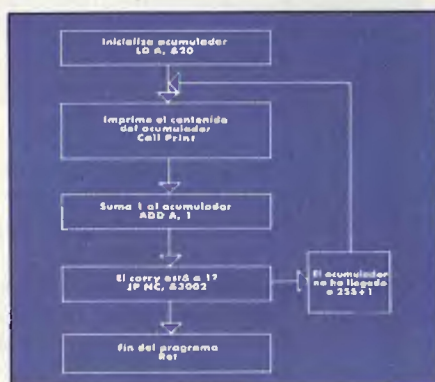
Instrucciones aritméticas: ADD A y SUB A.

Normalmente, existe más de una manera de hacer la misma cosa (al menos en el mundo del ordenador). Podemos incrementar el contenido del acumulador en una unidad con otra instrucción máquina, añadiendo flexibilidad a nuestros programas. Véase:

DIRECCION	CODIGO DE OPERACION	MNEMONICOS
3000	CD 18 BB	CALL ESPERAR
3003	C6 01	ADD A,1
3005	CD 5A BB	CALL PRINT
3008	C9	RET

El mnemónico ADD A, como su nombre indica, suma el byte que sigue a la coma (1 en este caso) al número contenido en el acumulador y DEJA EL RESULTADO EN EL ACUMULADOR.

Esta instrucción ocupa dos bytes: el propio código de operación (&C6) y el byte de datos que indica el número que hay que sumar al acumulador (1).



La ventaja de esta instrucción no queda clara en este ejemplo, ya que para hacer lo mismo que antes hemos ocupado un byte más de memoria. Ahora bien, lo interesante es que podemos sumar al acumulador cualquier número entre 0 y 255:

DIRECCION	CODIGO DE OPERACION	MNEMONICO
3000	3E 02	LD A,2
3002	C6 02	ADD A,2
3004	32 F8 2F	LD(&2FF8),A
3007	C9	RET

Usando la opción de examinar código en el Hexer, se verá que &2FF8 contiene 4, como esperábamos. Los números pequeños no causan problemas, las operaciones aritméticas con ellos funcionan según lo previsto.

Sin embargo, veamos que ocurre con los números grandes, como 255 (&FF):

DIRECCION	CODIGO DE OPERACION	MNEMONICOS
3000	3E FF	LD A,&FF
3002	C6 01	ADD A,1
3004	32 F8 2F	LD (&2FF8),A
3007	C9	RET

Al examinar como antes la posición de memoria &2FF8, resulta que contiene un flamante 0, en lugar del esperado 256.

¿Qué ha sucedido?

El problema estriba en que el acumulador sólo puede almacenar un byte a un tiempo, y $255 + 1$, si lo escribimos en binario (*mediante el comando Basic BIN\$*) veremos que necesita 9 BITS (*dos bytes*) para poderlo representar. De manera que el Z80 cuando debe incrementar 255 «redondea» a 0.

En lo relativo al acumulador:

$$\begin{aligned} 255 + 1 &= 0 \\ 255 + 2 &= 1 \\ 255 + 3 &= 2 \end{aligned}$$

Este comportamiento es muy similar al de un cuentakilómetros, cuando llega a 99.999 el siguiente paso es 0 kilómetros.

Resulta conveniente visualizar los números binarios como si se tratara de un círculo (Figura 1). Sumando avanzamos por el círculo en el sentido de las agujas del reloj, restando avanzamos en sentido contrario (*recuérdese una definición de byte «número binario que varía CICLICAMENTE entre 0 y 255*).

El siguiente programa muestra lo que sucede al restar:

DIRECCION	CODIGO DE OPERACION	MNEMONICO
300	3E 00	LD A,0
3002	D6 01	SUB A,01
3002	32 F8 2F	LD (&2FF),A
3007	C9	RET

La posición &2FF8, como ahora sí esperamos, contiene 255 (en decimal).

Se ha empleado una nueva instrucción, mnemónico SUB A,N, que sirve para restar el número N del acumulador y cuyo código de operación es &D6, seguido del número que restamos. El resultado de la operación QUEDA EN EL ACUMULADOR.

Lo que ocurre en los dos programas anteriores, al manipular el número 255, se conoce como SOBREPASAMIENTO, y puede dar lugar a serios problemas en nuestros programas si no se le presta atención.

Los flags son clave para que un programa tome decisiones

No obstante, el Z80 nos avisa con toda cortesía de que ha tenido lugar sobrepasamiento mediante lo que en inglés se conoce como «carry flag», y que en nuestro idioma podría traducirse como «bandera de acarreo» o «indicador de acarreo». Nosotros emplearemos, sin embargo, las palabras inglesas, porque aparecen en bastantes libros y, sobre todo, porque la inmensa mayoría de los programadores las usan. Así, hablaremos del CARRY FLAG o simplemente del CARRY.

Los FLAGS son muy importantes para los programadores: permiten chequear a lo largo de un programa cómo van las cosas, y se ven afectados por multitud de instrucciones máquina, especialmente las intrucciones de bifurcación o de salto según se cumpla o no una condición.

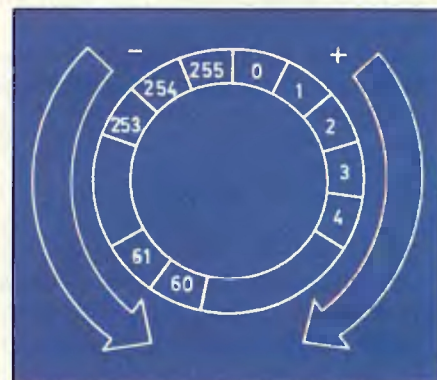


Figura 1: Presentación circular de los valores posibles de un byte.

Obviamente, un flag no es un termómetro. Sólo puede tomar dos valores distintos: 0 ó 1 (en el más puro estilo binario).

Cuando un flag está a 1 decimos que está activado (del inglés SET), y cuando esta a cero decimos que está desactivado (aunque los ingleses dicen CLEAR, borrado).

Pues bien, cuando sumamos 1 a 255, o restamos uno de 0, el carry se activa para indicarnos que algo extraño está pasando.

De hecho, cada vez que el Z80 ejecuta un SUB A o un ADD A, el carry se ajusta de acuerdo al resultado de la operación: si cae dentro de los «límites permitidos», el carry queda desactivado, a cero. De lo contrario, se pone a 1.

Dependiendo de que se dé o no esta circunstancia, podemos tomar dos cursos de acción distintos, esto es, BLIFURCAR EL PROGRAMA.

Como ya habréis podido observar, esto es muy parecido a las sentencias Basic IF...THEN.

Existen, como veremos más tarde, instrucciones máquina que toman automáticamente en cuenta el estado de los flags.

Salto absoluto: un primer paso.

Supongamos que introducimos mediante el Hexer el siguiente programa:

DIRECCION	CODIGO DE OPERACION	MNEMONICO
3000	C3 00 30	JP &3000

Este programa es bastante estúpido, equivale al programa Basic:

10 GOTO 10

sólo que si lo ejecutamos en lenguaje máquina, tendremos que desconectar el Amstrad para recuperar el control.

La instrucción JP le dice al micro que «salte» a la dirección de allí la ejecución del programa. Por tanto, el Z80 cumple la orden de saltar a la dirección &3000 y ejecutar lo que se encuentre allí, que es justamente JP &3000 de nuevo: un bucle sin fin.

Este tipo de salto se denomina SALTO INCONDICIONAL, PORQUE el ordenador obedecerá en cuanto llegue a la dirección donde JP se alla, independientemente de todas las condiciones o cambios de cualquier tipo que hayan podido tener lugar en el programa completo.

Existe además un grupo de instrucciones que permiten al Z80 «elegir», las llamadas de SALTO CONDICIONAL, y son las que en definitiva dotan a un programa de inteligencia.

La única que en este momento nos interesa es JP NC (del inglés Jump if Not Carry, salto si el carry flag está desactivado), cuyo código de operación es &D2.

Podemos aprovechar esta orden para crear un bucle, es decir, repetiremos un determinado grupo de instrucciones hasta que una condición se cumpla:

DIRECCION	CODIGO DE OPERACION	MNEMONICO
3000	3E 20	LD A,&20
3002	CD 5A BB	CALL PRINT
3005	C6 01	ADD A,1
3007	D2 02 30	JP NC,&3002
300A	C9	RET

La lógica de los bucles

Vamos a estudiar este programa con algo de detalle, porque encierra dos conceptos esenciales de la programación, tanto en código máquina como en general:

1. ITERACION: repetimos un grupo de órdenes cierta cantidad de veces hasta que...

2. LA CONDICION DE SALIDA DEL BUCLE SE CUMPLE.

En estos dos puntos, complementarios uno del otro, se pueden encuadrar el 90 por 100 de los programas.

Lo primero que hacemos es cargar el registro acumulador con el número &20, esto es, el código ASCII para el espacio.

A continuación, llamamos a la rutina del firmware etiquetada como PRINT (&BB5A), que imprime en pantalla el carácter contenido en el acumulador, preservando además este valor cuando la rutina acaba.

Luego sumamos 1 al acumulador, con lo que éste contendrá ahora

Código máquina

&21, el código ASCII correspondiente a «!».

En este momento llegamos a la instrucción de salto condicional, que hace lo siguiente:

a) SI EL CARRY FLAG ESTA A 1, ACTIVADO, se ejecuta la instrucción siguiente, es decir, RET, con lo que el programa acaba.

b) SI EL CARRY ESTA A 0, DESACTIVADO, la instrucción de salto condicional JP NC manda el control del programa a la dirección &3002, puesto que la condición se cumple.

Como a estas alturas el acumulador contiene &21, menor que 255, el carry ESTA DESACTIVADO. Por tanto, el bucle se repite: el carácter del acumulador se imprime en pantalla, su contenido se aumenta en una unidad y se vuelve a chequear el estado del carry.

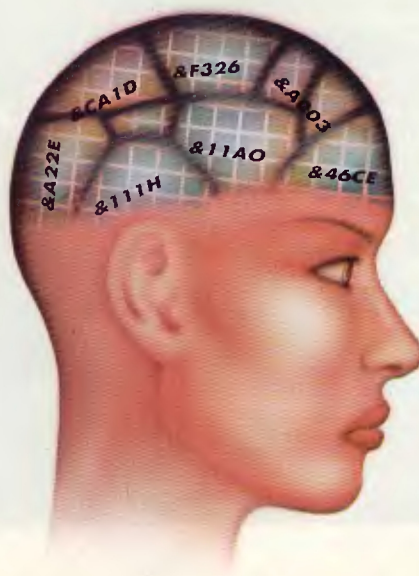
Ya resultará claro que este programa imprime todo el juego de caracteres del Amstrad a partir del espacio, porque al llegar el acumulador a contener 255, último carácter, e incrementarlo, el carry SE ACTIVA porque pasamos de 255 a 9, y la condición JP NC NO SE CUMPLE.

En el caso de que alguien se pregunte por qué no se ha empleado la instrucción INC A que ocupa un solo byte en lugar de ADD A, la razón es que aquella no AFECTA PARA NADA AL CARRY, con lo que el tipo de condición elegida no sirve.

Hemos aprendido cosas importantes esta semana:

Nuevas instrucciones en lenguaje máquina, y, sobre todo, la importancia y lógica de los bucles.

La próxima semana seguiremos estudiando otros registros de Z80.



PALABRAS SIMETRICAS

Análisis

Una palabra simétrica, es aquella que se deletrea de la misma forma comenzando por la izquierda que comenzando por la derecha. Palabras como ROTOR, RODADOR, RADAR, OJO, ORO, etc. responden a la definición de palabra simétrica.

En el análisis de esta semana, utilizamos las técnicas de fragmentación y manipulación de cadenas alfanuméricas para averiguar si una determinada palabra posee la característica de ser simétrica.

Esto se consigue con el uso de los comandos: MID\$ y LEN, que constituyen las dos órdenes básicas para trabajar con cadenas de caracteres.

10-20. Líneas que dan información a las personas que leen el listado. Son completamente ignoradas por el ordenador.

30-130. Constituye un bucle WHILE...WEND que no tiene fin. Lo cual se consigue gracias al NOT introducido entre el WHILE y la variable **terminar**.

Al no estar, toma automáticamente el valor cero, que en términos de lógica del ordenador equivale a falso.

El NOT, hace que el valor se invierta, provocando que éste se convierta en uno (verdadero), como el valor de **terminar** no es alterado por el programa, el bucle continúa infinitamente.

40. Nos pide una palabra, la cual queda almacenada en la variable alfanumérica **palabra\$**.

50. UPPER\$ pone la palabra en letras mayúsculas, con el objeto de evitar, los errores que se producirían al escribir una palabra que contenga letras mayúsculas y minúsculas a la vez, por ejemplo Rodador.

60. Calcula la longitud de la palabra a analizar, y la almacena en **long.palabra**.

70. Inicializa la variable **simétrica\$**, asignándole la cadena vacía.

80-110. Constituye un bucle FOR...NEXT, con tantos ciclos como letras contiene la palabra a analizar.

90. Cada vez que se repite el ciclo MID\$ toma una letra de la variable **palabra\$** y la almacena en **texto\$**.

Debido a que **letra** varía desde 1 hasta la

longitud total de la palabra, el proceso se repite, hasta que todas las letras son incluidas en **texto\$**.

100. **Texto\$** es añadido al principio de **simétrica\$**.

Cuando el bucle FOR...NEX, ha terminado de copiar las letras de **palabra\$** delante de **simétrica\$**, **simétrica\$** será la imagen de espejo de **palabra\$**.

120. Compara entre sí las variables **palabra\$** y **simétrica\$**. Si éstas son iguales implica que la palabra analizada es simétrica, enviando entonces el mensaje «ES UNA PALABRA SIMETRICA», en caso de que las variables comparadas no sean iguales, aparece en pantalla el mensaje colocado detrás del ELSE. «NO ES UNA PALABRA SIMETRICA.»



Siempre los primeros en tener lo mejor...

circulo de soft

MICROAMIGO S.A.



AJEDREZ (3 Dimensiones) 3.500 Ptas.
THE COVENANT 2.700 Ptas.
EXPLODING FIST 3.100 Ptas.

¡Un increíble regalo por cada programa!

Este magnífico reloj digital de cinco funciones puede ser tuyo si pides tus programas al **Círculo de Soft**. Si tu compra es de dos programas te obsequiamos con: **Un conmutador TV-Ordenador...** y ambos regalos si pides tres programas.



Nuestra dirección:
 Castellana, 268, 3.º C
 28046 MADRID
 Tel.: 733 25 00

y además...

GREMLINS	2.300 Ptas.
BEACH HEAD	2.700 Ptas.
COMBAT LINX	2.100 Ptas.
POLE POSITION	1.900 Ptas.
DRAGONTORE (prox.)	2.100 Ptas.
DECATHLON	2.100 Ptas.
KNIGH LORE (Novedad)	1.950 Ptas.
ALIEN 8 (Novedad)	1.950 Ptas.
KONG STRIKES BACK	2.100 Ptas.
HUNCHBACK II	2.100 Ptas.
ONE ON ONE (Novedad) ..	3.100 Ptas.
ROCKY	2.100 Ptas.
RAID OVER MOSCOW	2.100 Ptas.
BOXING	1.950 Ptas.

CUPON DE PEDIDO

Recorta o copia este cupón, o pide tus programas por teléfono. Deseo recibir a vuelta de correo el/los siguiente(s) programá(s).

TÍTULO

☐ Contrarreembolso ☐ Giro Postal ☐ Talón adjunto a Microamigo, S. A.

Nombre

Apellidos

Domicilio

Localidad

Provincia

Teléfono

P.V.P.

Fecha caducidad

C.P.

Edad

ORDENADOR

LA PANTALLA GRAFICA: ARTE EN ALTA RESOLUCION

Habiendo realizado en anteriores artículos un estudio acerca de la pantalla de texto y sus ventanas asociadas, podemos ahora centrar nuestra atención en la pantalla de gráficos y sus múltiples aplicaciones.

Al igual que ocurre con la pantalla de texto, al conectar su ordenador, la pantalla de gráficos ocupará la totalidad del área de éste (excepto, por supuesto, la zona del borde), y podremos trabajar en cualquiera de los tres modos diferentes que su AMSTRAD posee.

La pantalla gráfica usa sus propias coordenadas

Lo primero con lo que necesitamos familiarizarnos es con el sistema de coordenadas utilizado por la pantalla de gráficos, que es como se muestra en la Fig. 1. Como vemos, éstas difieren bastante de las utilizadas por la pantalla de texto, con el fin de permitirnos una mayor precisión a la hora de especificar puntos en la pantalla, tarea ésta mucho más complicada que la de simplemente colocar caracteres en un texto. Por esta razón para trabajar con gráficos dividimos la pantalla en una red de 640X400.

Para determinar un punto en el área de la pantalla de gráficos, introduciremos sus coordenadas especificando primero posición horizontal y después su posición vertical, tal como lo aprendimos en la escuela.

Debemos darnos cuenta de que el origen está situado en el extremo inferior izquierdo y que corresponde a las coordenadas (0,0). El eje X (horizontal) varía de 0 a 639 y el eje Y (vertical) de 0 a 399.

Esto funciona de una manera totalmente dis-

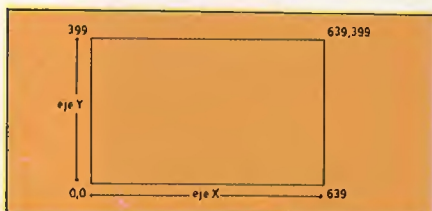


Figura 1: Coordenadas gráficas de la pantalla.

tinta a como lo hace la pantalla de texto, en la cual el origen está situado en el extremo superior izquierdo y la numeración comienza en 1.

En una ampliación de la pantalla de gráficos, cualquiera que sea la modalidad con la que estemos trabajando, la forma y la localización física de cualquier figura serán siempre las mismas. Por ejemplo el punto (0,0), estará siempre el extremo inferior izquierdo y el punto (639, 399) en el superior derecho.

Los comandos DRAW y MOVE

Intentemos dibujar algunas líneas en la pantalla. Para esto imaginemos algo llamado el cursor de gráficos, el cual es invisible aún, pero ocupa una localización específica de la pantalla.

Inicialmente cuando conectamos el ordenador, el cursor estará siempre en el origen de coordenadas (0,0). Podemos usar el comando DRAW para dibujar una línea desde la posición actual del cursor hasta cualquier otro punto de la pantalla. Por ejemplo si introducimos el comando:

DRAW 639,399

una línea será trazada diagonalmente a través de la pantalla desde la posición (0,0) hasta la (639,399).

También podemos colocar el cursor en cualquier punto sin tener que trazar una línea. El comando que realiza esta operación es MOVE y utiliza el mismo sistema de coordenadas que el anterior. Así pues podemos dibujar una recta desde la posición de cursor a cualquier otra, o bien simplemente moverlo hacia otro punto.

MOVE se usa igual que el comando DRAW, es decir, especificando las coordenadas X e Y. Así el comando:

MOVE 639,399

moverá el cursor a través de la pantalla hacia la esquina superior derecha, pero esta vez no se dibujará ninguna línea.

Ahora realizaremos unos programas simples para poner en práctica estas teorías. El programa 1 traza nuestra diagonal de nuevo, se-

guida por una segunda línea hacia abajo por el borde derecho y finalmente una tercera que vuelve el punto inicial formando un triángulo.

A propósito, hemos incluido en este programa un «dummy» en la última línea para prevenir al cursor de texto y el mensaje «Ready» aparecerá en la pantalla una vez se halla completado el dibujo. Esto ayuda a mantener ocupada a la pantalla de gráficos para este tipo de programas de demostración. Por supuesto que tendremos que presionar «Escape» dos veces para salir del programa.

Ahora para ilustrar el uso del comando MOVE, cambia la línea 40 por:

40 MOVE 639,8

si ahora corre el programa verá que el trián-



mos intentado hacer alguna vez con un lápiz y un papel pero que ahora podemos intentar con un ordenador. Le prometemos que se puede hacer.

Hacerlo de otro modo

Los programas I y II han utilizado el «Mode» 1 para realizar los ejemplos, pero también podemos utilizar los «Modes» 0 y 2. Recordará que cuando estuvimos practicando con la pantalla de texto la modalidad controlaba el número de caracteres que podíamos imprimir en la pantalla. El comando «Mode» también tiene un efecto en la pantalla de gráficos pero de una manera muy diferente. Cambie la línea 20 del programa II por:

20 MODE 0

y ejecute de nuevo el programa. Como verá, la figura se ha dibujado de la misma manera y ocupa la misma posición y área que la anterior. Sin embargo, se habrá dado cuenta que las líneas aparecen ligeramente más gruesas que con el «Mode» 1 y que las diagonales tienen una apariencia de escalera en lugar de una línea recta más suave.

Esto es debido a que la modalidad 0 utiliza una resolución de gráficos más débil, y los resultados son más bastos que con la modalidad

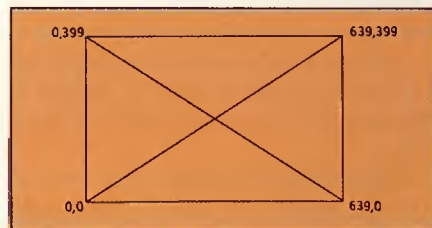


Figura 2: Resultado del programa 2.

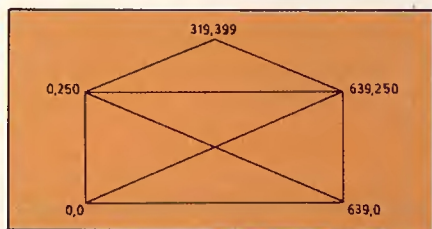


Figura 3: Resultado del programa 4.

gulo queda incompleto, pues en lugar de dibujar una línea en la parte derecha, tan sólo hemos movido el cursor a través de ella.

Intentamos un ejemplo más complicado. El programa II traza un rectángulo con sus dos diagonales como muestra la Fig. II. Vea cómo hemos utilizado los comandos MOVE y DRAW para realizar esta figura.

Por supuesto que éste no se podría realizar utilizando únicamente el comando DRAW (a no ser que dibujáramos unas líneas encima de otras), así pues necesitaremos la ayuda del comando MOVE.

Un buen ejercicio. Intente reproducir el dibujo del sobre de la Fig. III usando solamente comandos DRAW. Esto es algo que todos he-

1. Cuando utilizamos un «Mode» más potente los puntos son mucho más pequeños y las líneas más finas.

Si queremos utilizar una resolución de gráficos realmente alta, podemos utilizar el «Mode» 2. Inténtelo con el programa II y vea los efectos de su uso. Notará cómo nos permite crear dibujos con más lujo de detalles, pero como cabría esperar, esta ventaja debemos pagarla con una menor calidad del color.

¿Recuerda cómo la modalidad restringía al uso del color en la pantalla de texto? Por desgracia las mismas reglas tienen lugar en la pantalla de gráficos.

En la resolución de gráficos del «Mode» 2 sólo es posible utilizar dos colores. Además si

Program Acción

utilizamos uno para el fondo, sólo podremos dibujar líneas en otro color. Por supuesto que podemos elegir entre 27 colores posibles tal y como se realiza con el comando INK.

El «Mode» 0 puede utilizarse para dibujos de hasta 16 colores, pero perdemos calidad en el trazado de las líneas. Así pues será usted el que tenga que escoger la opción que más le convenga.

Para medir la resolución de la pantalla se utilizan los llamados pixels: cada uno es un punto de color aislado.

La tabla I muestra la relación entre los resultados de la pantalla. Podemos ver que la resolución vertical siempre es 200, mientras que la horizontal varía según la modalidad.

Hay que aclarar que un pixel no corresponde necesariamente a un punto de la pantalla, pues en la mayoría de las ocasiones puntos vecinos tienen que compartir el mismo pixel.

Lo más importante que debemos recordar es que las coordenadas usadas en los comandos DRAW y MOVE son siempre de 0 a 639 horizontalmente y de 0 a 399 verticalmente, indistintamente de la modalidad elegida.

Esto puede parecer un poco complicado al principio, pero comprobará que resulta más fácil trabajar usando sólo un sistema de coordenadas en lugar de los tres mostrados en la tabla I.

Para asegurarnos que ha entendido los efectos de las diferentes modalidades, introduzca el programa III. Este utiliza las tres posibilidades y dibuja una línea al centro de la pantalla; en lo que debe fijarse es en que siempre se utilizan las mismas coordenadas y que el grosor de la línea varía para cada una de ellas. La línea 60 simplemente introduce un pequeño retraso entre cada opción.

Podemos ver que nuestras líneas han sido dibujadas en amarillo brillante sobre fondo azul. Estos son los colores que el ordenador utiliza cuando le conectamos. Intentemos realizar un cambio de color en nuestra línea dibujada.

Dibujando a todo color

En los ejemplos anteriores no especificamos ningún color, y por tanto el ordenador entendió que queríamos simplemente dibujar sin importarnos el color. Si nosotros no queremos que esto sea así, podremos cambiar éstos añadiendo un tercer parámetro al comando DRAW. Introduzca los siguientes comandos:

CLG
DRAW 639,399,2

ahora nuestra diagonal aparecerá en un brillante azul claro. Como ve, el tercer número del comando DRAW le dice qué tinta utilizar. Una vez ejecutado este comando, todas las líneas aparecerán en dicho color, hasta que especifiquemos una nueva tinta con otro comando DRAW.

Esto también ocurre con las tintas de los textos, pues una vez elegidos permanece su uso hasta que aparezca un nuevo comando PEN. Lo verá si teclea:

DRAW 639,0

la segunda línea también se dibujará en azul claro, a pesar de que ahora no hemos especificado ningún color.

Ahora introduzca el comando:

DRAW 0,0,1

y el color cambiará a amarillo para completar el triángulo.

El programa IV dibuja el sobre con el que anteriormente habrá estado luchando. Esta vez estamos usando el 'Mode'0, por tanto, podemos utilizar una gran variedad de colores para nuestro ejemplo. Podrá ver que cada línea está dibujada en un color diferente, pues hemos introducido un tercer parámetro para cada comando DRAW.

Pruebe a omitir algunos de estos parámetros en este programa y verá cómo se comportan los colores del dibujo.

Siguiendo con el tema del color, veamos cómo podemos variar el color de fondo en nuestra pantalla.

El comando CLG se usa para limpiar la pantalla de gráficos, y funciona de una manera muy similar al comando CLS, utilizado para borrar la pantalla de texto.

Sin embargo, si utiliza CLG directamente, es decir, teclearlo en su **Amstrad** y después presionar Enter, aparecerá el mensaje «Ready» en la línea correspondiente a la última posición del cursor de texto y no en la parte superior de la pantalla. También podemos añadirle a este comando un número de tinta con el fin de cambiar el color de fondo.

Teclee RESET y después:

CLG 3

como la pantalla estaba limpia, ha cambiado al color especificado, es decir, al rojo. Puede elegirse cualquier color siempre y cuando esté incluido dentro del rango permitido por la modalidad utilizada. Esta acción es muy similar a la de cambiar el color del papel en la pantalla de texto, pero debido a que en este caso se debe limpiar la pantalla para cambiar de color, ésta lo hará en su totalidad y de una vez.

La pantalla gráfica puede tomar cualquier color de fondo

El programa V demuestra cómo se puede cambiar el color de fondo con el comando CLG. Dese cuenta que el papel del texto no

es afectado por este comando y de que el cursor de texto no vuelve a su posición inicial como lo hacía con el comando CLS.

Sería interesante que intentara cambiar los colores de fondo en alguno de los programas anteriores. Una advertencia: Tenga cuidado de no dibujar líneas sobre el fondo del mismo color; estas líneas serán dibujadas, pero no las podrá ver.

Introduciendo elementos aleatorios en los dibujos se pueden obtener efectos muy interesantes, por ejemplo, utilizando colores y coordenadas elegidos al azar.

En el programa VI se dibujan unas líneas manejando este concepto. Después de ser trazada cada línea, se lleva el cursor al origen de coordenadas y el resultado final es una serie de rectas que irradian desde el extremo inferior izquierdo.

En el caso de que emitiéramos la línea 80, el cursor no volvería al origen. ¿Qué cree usted que ocurriría? Pruébalo y lo verá.

También podríamos cambiar el cursor de posición sin tener necesariamente que volver al origen. Haga varias pruebas con estas opciones y corra el programa de nuevo.

Otro efecto interesante se puede conseguir cambiando las funciones aleatorias por otras más lógicas. El programa VII de nuevo hace uso de los comandos DRAW y MOVE, pero esta vez las líneas son dibujadas de un modo más ordenado.

Empezamos con dos puntos fijos (X, Y) y (a, b) en los extremos de la pantalla y trazamos una línea entre ellos. Entonces se cambian los valores de las cuatro variables en pasos de 8 unidades antes de que la siguiente recta se dibuje.

Existen muchas variaciones sobre el mismo tema. El programa VIII es otro ejemplo. Intente escribir algunos programas similares a éstos y advine qué aspecto tendrán antes de correrlos.

Con el uso de los comandos DRAW y MOVE se pueden obtener muchos efectos creativos limitados solamente por su propia imaginación.

Figuras dibujadas con respecto a un punto

Ahora que ya se habrá familiarizado con el sistema de coordenadas y con estos comandos, veamos el uso de otros dos nuevos que presentan unas leves variaciones con respecto a los anteriores. Estos son MOVE y DRAW. La R que se les añade al final es la inicial de la palabra «Relativo», y simplemente significa que el movimiento o el trazado ven-

drán dados en relación a la posición anterior del cursor.

Comencemos limpiando la pantalla, así sabremos que el cursor estará situado en el origen. Si tecleamos ahora:

DRAW 200,200
DRAW 400,0

obtendremos una especie de V invertida, pero debe darse cuenta de que todas las coordenadas usadas hasta ahora vienen dadas con referencia al punto (0,0). Estas son las llamadas coordenadas absolutas.

Realicemos ahora el mismo ejemplo usando los nuevos comandos. Antes teclee CLG para situar el cursor en el origen y luego, los siguientes comandos:

DRAWR 200,200
DRAWR 400,0

Esta vez las líneas han sido trazadas con relación a la última posición del cursor. La primera línea es idéntica a la del caso anterior, pues el cursor también se encontraba en el origen. Sin embargo, la segunda línea es completamente distinta, pues ha sido dibujada horizontalmente desde el final de la recta anterior y acaba en un punto de coordenadas absolutas (600,200). Lo que ha ocurrido es que esta recta ha sido trazada con relación a la última posición del cursor en lugar desde el punto de origen. El hecho es que las dos coordenadas han sido sumadas de la siguiente manera:

$$200 + 400 = 600$$

$$200 + 0 = 200$$

y por esto hemos finalizado en el punto (600,200). Las mismas reglas se aplican al usar el comando MOVE. Si tecleamos los comandos:

CLG
MOVER 150,200
MOVER 100,150

y nuestras matemáticas son correctas, el cursor se encontrará en el punto (250,350).

Como verá el uso de las coordenadas relativas es tan sencillo como el de las absolutas. El único problema es que puede salirse fácilmente del borde de la pantalla cuando esté trabajando con coordenadas relativas, por tanto, deberá tener cuidado de dónde está situado el cursor en cada momento.

Pruebe a cambiar los comandos DRAW y MOVE de algunos de los programas realizados hasta ahora por DRAWR y MOVER. Si no varían las coordenadas, las figuras resultantes serán totalmente diferentes a las originales, pero seguro que es capaz de ajustar dichos valores para que se obtengan los mismos resultados.

En teoría cualquier figura puede ser dibujada utilizando tanto el sistema de coordenadas relativas como el absoluto.

Es interesante aclarar que se permite utilizar coordenadas negativas, así los comandos:

CLG
MOVER 200,200
MOVER -200,-200

Mode Number	Horizontal Pixels	Vertical Pixels
0	160	200
1	320	200
2	640	200

Tabla 1: Resolución gráfica según el modo.

llevarían el cursor al punto 200,200 y volvería de nuevo al origen. De hecho, los números negativos son casi imprescindibles para dibujar con coordenadas relativas.

El programa IX demuestra este punto, dibujando un cuadro en dicho sistema.

Otro detalle de interés es que debe considerarse que es posible sacar el cursor fuera de la pantalla usando coordenadas mayores que las normalmente permitidas.

Hasta ahora sólo hemos utilizado valores incluidos dentro del rango (639,399), pero también es válido e incluso a veces necesario manejar números mayores que éstos.

Introduzca los comandos:

```
CLG
DRAW 1000,1000
```

y su ordenador trazará sin problemas una línea hasta ese punto imaginario. Por supuesto que la línea desaparecerá de la pantalla. Para comprobar que efectivamente el cursor ha ido a ese punto teclee:

```
DRAW 500,0
```

y verá cómo reaparece hasta el punto especificado como si realmente viniera del (1000,1000).

El programa X demuestra cómo pueden ser utilizadas las coordenadas del exterior de la pantalla. Comienza dibujando un triángulo lo suficientemente pequeño como para caber en la pantalla. Si ahora presiona la tecla el triángulo se borrará y reaparecerá ligeramente más grande. Siga presionando dicha tecla y poco a poco el triángulo irá desapareciendo por los extremos de la pantalla. Lo que ocurre es que está siendo dibujado en coordenadas fuera de la misma. Utilice la tecla —para que la figura vuelva a su tamaño inicial—. Esta técnica, llamada de «ZOOM», se utiliza frecuentemente para mostrar ciertas imágenes en dibujos complejos.

Y con esto ya está bien por este mes. Intente realizar dibujos de su propia invención utilizando los comandos DRAW, MOVE, DRAWR y MOVER, y disfrute de el placer de dibujar con su ordenador.

```
10 REM PROGRAMA I
20 MODE 1
30 DRAW 639,399
40 DRAW 639,0
50 DRAW 0,0
60 GOTO 60
```

```
10 REM PROGRAMA II
20 MODE 1
30 DRAW 639,399
40 DRAW 639,0
50 DRAW 0,0
60 DRAW 0,399
70 DRAW 639,0
80 MOVE 0,399
90 DRAW 639,399
100 GOTO 100
```

```
10 REM PROGRAMA III
20 FOR resolucion= 0 TO 2
30 MODE resolucion
40 PRINT "mode ";resolucion;
50 DRAW 320,200
60 FOR pausa=1 TO 2000:NEXT
70 NEXT resolucion
80 GOTO 20
```

```
10 REM PROGRAMA IV
20 MODE 0
30 DRAW 639,0,2
40 DRAW 639,250,3
50 DRAW 0,0,4
60 DRAW 0,250,5
70 DRAW 639,250,6
80 DRAW 319,399,7
90 DRAW 0,250,8
100 DRAW 639,0,9
110 GOTO 110
```

```
10 REM PROGRAMA V
20 MODE 0
30 FOR color=0 TO 15
40 CLG color
50 PRINT "CLG ";color
60 FOR pausa=1 TO 1000:NEXT
70 NEXT
```

```
10 REM PROGRAMA VI
20 MODE 0
30 WHILE INKEY$=""
40 color=INT(15*RND(1))+1
50 x=INT(639*RND(1))+1
60 y=INT(399*RND(1))+1
70 MOVE 0,0
80 DRAW x,y,color
90 WEND
```

```
10 REM PROGRAMA VII
20 MODE 0
30 FOR color=1 TO 15
40 x=0
50 y=0
60 a=639
70 b=399
80 WHILE y<400 AND x<640
90 MOVE x,y
100 DRAW a,b,color
110 x=x+8
120 y=y+8
130 a=a-8
140 b=b-8
150 WEND
160 NEXT color
```

```
10 REM PROGRAMA VIII
20 MODE 0
30 x=0:y=200
40 a=639:b=200
50 WHILE a>320
60 MOVE x,y
70 DRAW a,b,2
80 a=a-8:b=b-4
90 WEND
100 WHILE x<320
110 MOVE x,y
120 DRAW a,b,3
130 x=x+8:y=y+4
140 WEND
150 WHILE a>0
160 MOVE x,y
170 DRAW a,b,4
180 a=a-8:b=b+4
190 WEND
200 WHILE x<639
210 MOVE x,y
220 DRAW a,b,5
230 x=x+8:y=y-4
240 WEND
250 WHILE a<320
260 MOVE x,y
270 DRAW a,b,6
280 a=a+8:b=b+4
290 WEND
300 WHILE x>320
310 MOVE x,y
320 DRAW a,b,7
330 x=x-8:y=y-4
340 WEND
350 WHILE a<640
360 MOVE x,y
370 DRAW a,b,8
380 a=a+8:b=b-4
390 WEND
400 WHILE x>0
410 MOVE x,y
420 DRAW a,b,9
430 x=x-8:y=y+4
440 WEND
450 GOTO 30
```

```
10 REM PROGRAMA IX
20 MODE 1
30 MOVE 300,300
40 DRAWR 200,0
50 DRAWR 0,-200
60 DRAWR -200,0
70 DRAWR 0,200
```

```
10 REM PROGRAMA X
20 MODE 1
30 color=1
40 x=100:y=100
50 a=400:b=300
60 c=400:d=100
70 GOSUB 110
80 IF NOT INKEY(28) THEN GOSUB 170
90 IF NOT INKEY(25) THEN GOSUB 250
100 GOTO 80
110 REM dibuja triangulo
120 MOVE x,y
130 DRAW a,b,color
140 DRAW c,d
150 DRAW x,y
160 RETURN
170 color=0
180 GOSUB 110
190 x=x-8:y=y-8
200 a=a+8:b=b+8
210 c=c+8:d=d-8
220 color=1
230 GOSUB 110
240 RETURN
250 color=0
260 GOSUB 110
270 x=x+8:y=y+8
280 a=a-8:b=b-8
290 c=c-8:d=d+8
300 color=1
310 GOSUB 110
320 RETURN
```



ASTRON 1400

IMPRESIONANTE!

Impresionante en precio y prestaciones. Con todo lo que puede y debe exigirse a una impresora de la más avanzada tecnología: alta velocidad (140 caracteres por segundo), amplio juego de caracteres, alimentación con cualquier tipo de papel (continuo, hojas sueltas, en rollo), excelente calidad de impresión (matriz de 9×9), interface centronics (100% compatible con el IBM PC) y, opcionalmente, Serie RS-232 C, ...

Compacta y robusta hasta en el menor detalle de diseño y a un precio que hasta hoy parecía imposible. Impresionante ASTRON 1400.

ESPECIFICACIONES

Sistema: Matriz de puntos (9×9) con agujas reemplazables.
Velocidad: 140 c.p.s.
Fuentes: 96 caracteres ASCII y 32 semi-gráficos.

Columnas: 80 (caracteres normales), 132 (comprimidos), 40 (ensanchados), 66 (ensanchados-comprimidos).

Alimentación: Fricción y tracción.

Papel: Hojas sueltas de 102 a 254 mm. de anchura.
 Papel continuo de 242 mm. (opcionalmente de 102 a 254 mm.).
 Papel en rollo de 102 a 254 mm.

Interface: Paralelo CENTRONICS. Opcionalmente, Serie RS-232C.

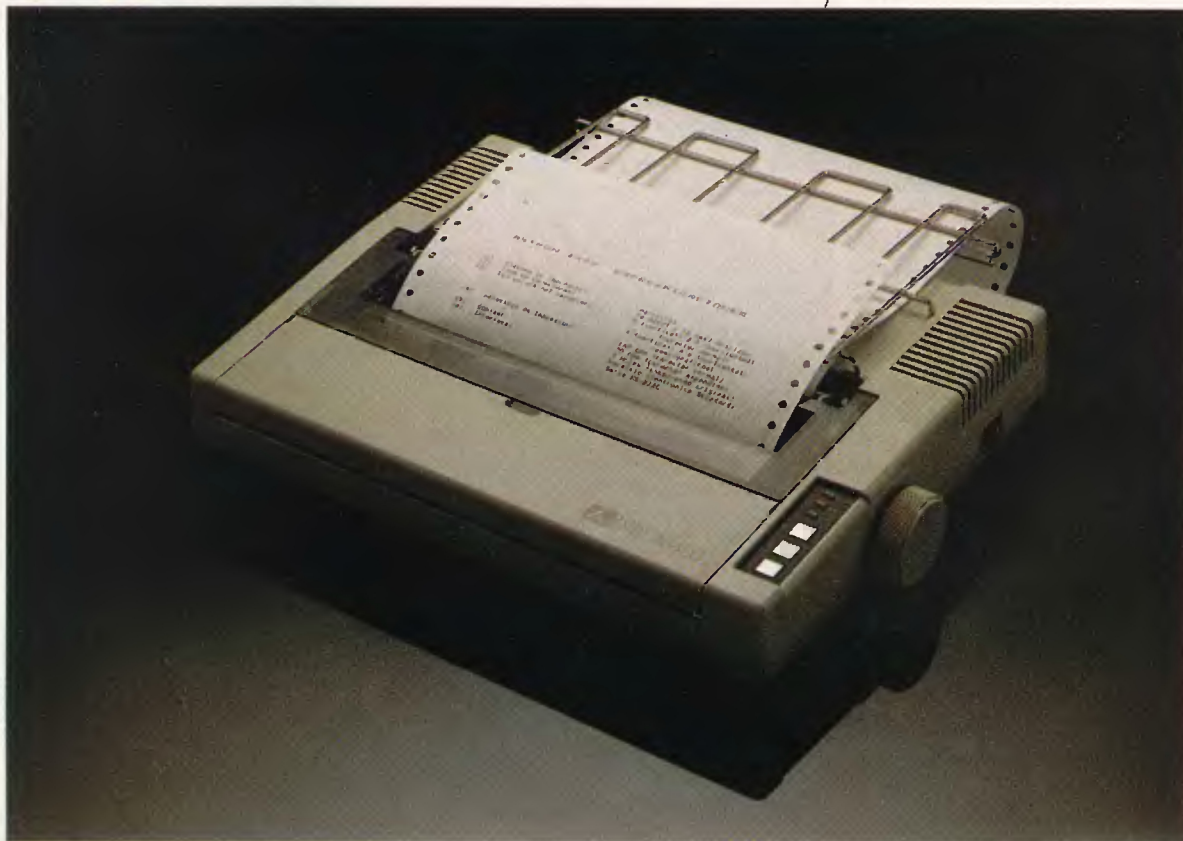
Buffer: 2 kbytes de memoria.

P.V.P.

66.900 pts.



Pº de la Castellana, 179. 28046 MADRID. Teléfono: 442 54 33./ 44



GANA 100.000 PESETAS CON MICROHOBBY AMSTRAD SEMANAL

Porque pretendemos que **AMSTRAD SEMANAL** sea también vuestra revista, hemos abierto una sección en la que se publicarán los mejores programas originales recibidos en nuestra redacción. Vosotros seréis los encargados de realizar estas páginas, en las que podréis aportar ideas y programas interesantes para otros lectores.

Las condiciones son sencillas:

- Los programas se enviarán a **AMSTRAD SEMANAL** en una cinta de cassette, sin protección en el software, de forma que sea posible obtener un listado de los mismos.
- Cada programa debe ir acompañado de un texto explicativo en el cual se incluyan:
 - Descripción general del programa.
 - Tabla de subrutinas y variables utilizadas, explicando claramente la función de cada una de ellas.
 - Instrucciones de manejo.

— Todos estos datos deberán ir escritos a máquina o con letra clara para mayor comprensión del programa.

— En una sola cinta puede introducirse más de un programa.

— Una vez publicado, **AMSTRAD SEMANAL** abonará al autor del programa de **15.000 a 100.000** pesetas, en concepto de derechos de autor.

— Los autores de los programas seleccionados para su publicación, recibirán una comunicación escrita de ello en un plazo no superior a dos meses a partir de la fecha en que su programa llegue a nuestra redacción.

— **AMSTRAD SEMANAL** se reserva el derecho de publicación o no del programa.

— Todos los programas recibidos quedarán en poder de **AMSTRAD SEMANAL**.

— Los programas sospechosos de plagio serán eliminados inmediatamente.

¡ENVÍANOS TU PROGRAMA!

a **HOBBY PRESS, S. A.** La Granja, s/n. Pol. Ind. Alcobendas (Madrid)

Mercado común

Con el objeto de fomentar las relaciones entre los usuarios de **AMSTRAD**, **MERCADO COMUN** te ofrece sus páginas para publicar los pequeños anuncios que relacionados con el ordenador y su mundo se ajusten al formato indicado a continuación.

En **MERCADO COMUN** tienen cabida, anuncios de ventas, compras, clubs de usuarios de **AMSTRAD**, programadores, y en general cualquier clase de anuncio que pueda servir de utilidad a nuestros lectores.

Envíanos tu anuncio mecanografiado a: **HOBBY PRESS, S.A.**

AMSTRAD SEMANAL.

Apartado de correos 54.062
28080 MADRID

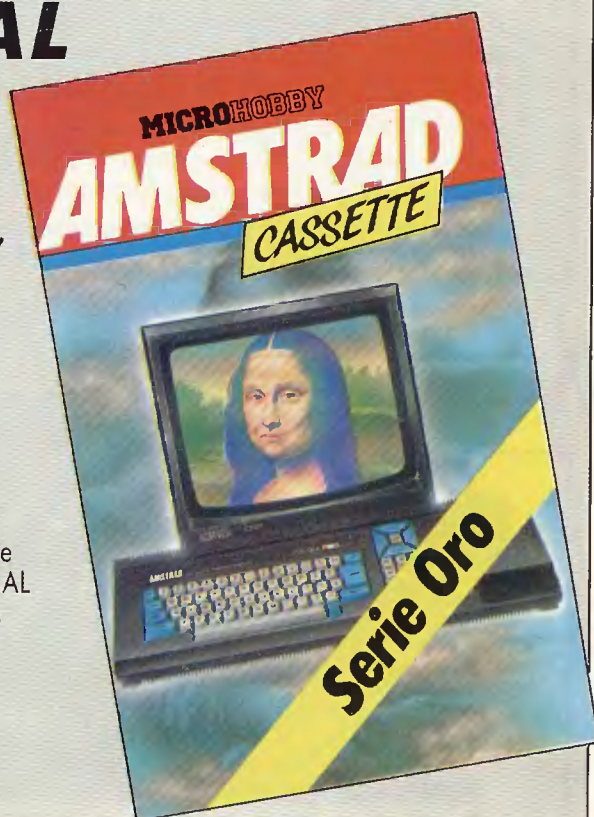
¡ABSTENERSE PIRATAS!

MICROHOBBY AMSTRAD SEMANAL

**LE OFRECE AHORA SUS
PROGRAMAS YA GRABADOS,
PARA QUE VD. NO TENGA
QUE TECLEARLOS**

Todos los programadores y aficionados a la microinformática sabemos lo tedioso y propenso a errores que resulta el teclear un listado de un programa. Para facilitar tu labor al máximo y que no tengas que estar horas sobre el teclado de tu ordenador tratando de descifrar incomprensibles mensajes de error, **AMSTRAD SEMANAL** te ofrece cada mes los programas publicados de los cuatro números correspondientes en una cinta de cassette, sólo por **675 ptas.** (sin más gastos por envío).

**Envíanos con la menor demora posible,
el cupón correspondiente.**



COM

A

P

Porque
MANAL se
abierto una
mejores pro
tra redacci
realizar est
tar ideas y
lectores.

Las con

— Los p
SEMANTAL
ción en el
obtener un

— Cada programa debe ir acompañado de
un texto explicativo en el cual se incluyan:

— Descripción general del programa.

— Tabla de subrutinas y variables utilizadas,
explicando claramente la función de cada
una de ellas.

— Instrucciones de manejo.

grama llegue a nuestra redacción.

— AMSTRAD SEMANTAL se reserva el de-
recho de publicación o no del programa.

— Todos los programas recibidos quedarán
en poder de AMSTRAD SEMANTAL.

— Los programas sospechosos de plagio se-
rán eliminados inmediatamente.

¡ENVÍANOS TU PROGRAMA!

a HOBBY PRESS, S. A. La Granja, s/n. Pol. Ind. Alcobendas (Madrid)

Franqueo
Postal

HOBBY PRESS, S.A.

Apartado de Correos

n.º 54.062 (Apartados Altos)

MADRID

que pueda servir de utilidad a nuestros
lectores.

Envíanos tu anuncio mecanografiado

a: **HOBBY PRESS, S.A.**

AMSTRAD SEMANTAL.

Apartado de correos 54.062
28080 MADRID

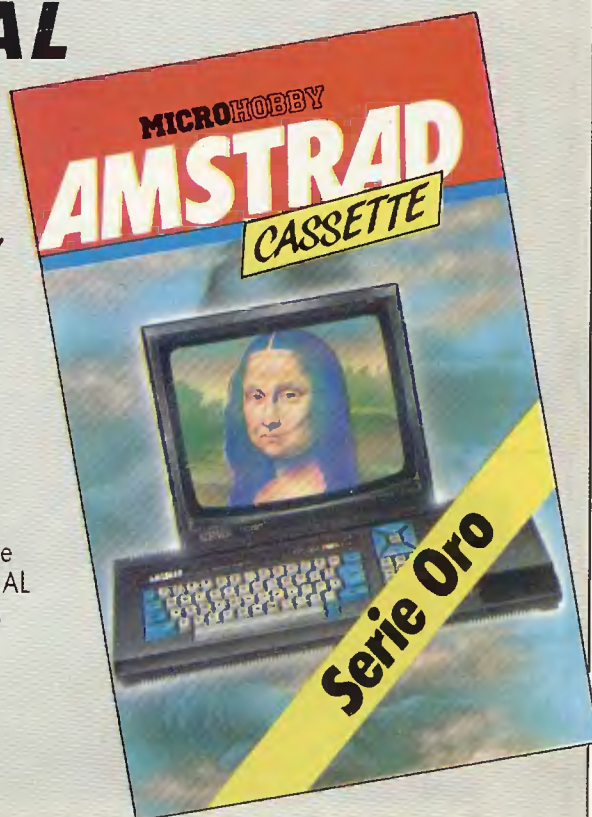
¡ABSTENERSE PIRATAS!

MICROHOBBY AMSTRAD SEMANTAL

**LE OFRECE AHORA SUS
PROGRAMAS YA GRABADOS,
PARA QUE VD. NO TENGA
QUE TECLEARLOS**

Todos los programadores y aficionados a la microinformática sabemos lo tedioso y propenso a errores que resulta el teclear un listado de un programa. Para facilitar tu labor al máximo y que no tengas que estar horas sobre el teclado de tu ordenador tratando de descifrar incomprensibles mensajes de error, **AMSTRAD SEMANTAL** te ofrece cada mes los programas publicados de los cuatro números correspondientes en una cinta de cassette, sólo por **675 ptas.** (sin más gastos por envío).

**Envíanos con la menor demora posible,
el cupón correspondiente.**



Contar por esta línea

¡¡SUSCRIBETE A Microhobby AMSTRAD Y AHORRA 1.600 PTAS.!!

SOLICITUD DE CINTAS DE PROGRAMAS Y NUMEROS ATRASADOS

☐ Deseo recibir en mi domicilio, al precio de **675 ptas.** cada una, las siguientes cintas con los programas publicados por **Microhobby AMSTRAD** Marco con una (X) la(s) cinta(s) que deseo:

- ☐ **Cinta n.º 1** (contiene programas publicados en revistas 1 al 4 inclusive)
- ☐ **Cinta n.º 2** (contiene programas publicados en revistas 5 al 8 inclusive)
- ☐ **Cinta n.º 3** (contiene programas publicados en revistas 9 al 12 inclusive)
- ☐ **Cinta n.º 4** (contiene programas publicados en revistas 13 al 16 inclusive)
- ☐ **Cinta n.º 5** (contiene programas publicados en revistas 17 al 20 inclusive)

☐ Deseo recibir en mi domicilio, al precio de **150 ptas.** cada uno, los siguientes números atrasados de **Microhobby AMSTRAD** _____

Nota: Por razones administrativas, no podemos admitir solicitudes de envío de cintas o números sueltos con pago contra reembolso o Tarjeta de Crédito. Por favor, envía talón por el importe o giro postal.
Si la forma de pago elegida es talón bancario, remítelo junto con este cupón en un sobre cerrado a la misma dirección.
Las cintas de programas se editan una cada mes. Si solicitas varias las recibirás sucesivamente, conforme sean editadas. No se cobran gastos de envío por las cintas y números sueltos.

NOMBRE _____ EDAD _____
 APELLIDOS _____
 DOMICILIO _____
 CIUDAD _____ PROVINCIA _____
 C. POSTAL _____ TELEFONO _____ PROFESION _____

Marco con una (X) en el casillero correspondiente la forma de pago que más me conviene.

☐ Talón bancario adjunto a nombre HOBBY PRESS, S. A. ☐ Giro Postal N.º _____

Firma: _____

Manual de Referencia Basic para el Programador
 La más autorizada y completa guía para programar en Locomotive Basic.
 3.400.— Pts.



Código máquina para principiantes con AMSTRAD
 Ideal para iniciarse en el código máquina del Z80 y en el sistema operativo del AMSTRAD.
 2.100.— Pts.



Técnicas de Programación de Gráficos en el AMSTRAD
 Este libro enseña a aprovechar las excelentes funciones gráficas del AMSTRAD, con múltiples ejemplos.
 1.950.— Pts.

Juegos Sensacionales para AMSTRAD
 Listados completos de 27 estupendos juegos de muy diversos estilos.
 1.950.— Pts.



Hacia la Inteligencia Artificial con AMSTRAD
 Convierta su AMSTRAD en un compañero inteligente
 1.500.— Pts.

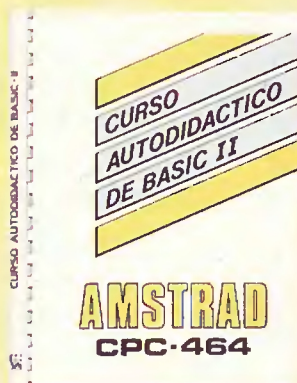


Curso Autodidáctico de Basic I y II
 Un completo y estructurado Curso de Basic apoyado con numerosos ejemplos y acompañado de cassettes.
 2.900.— Pts. cada volumen

Programando con AMSTRAD
 Fundamental para el usuario principiante.
 Ameno y repleto de ejemplos.
 2.400.— Pts.



Música y Sonidos con AMSTRAD
 Programe música y efectos sonoros y convierta su AMSTRAD en un sintetizador.
 1.200.— Pts.



40 Juegos Educativos
 Listados completos (matemáticas, geografía, música, etc.) para aprender divirtiéndose.
 1.950.— Pts.



Programación de Basic con AMSTRAD.
 Imprescindible para el principiante y eficaz herramienta para el programador avanzado.
 2.100.— Pts.

indescomp
 PUBLICACIONES

Avd. del Mediterráneo, 9
 Telfs.: 433 45 48 — 433 48 76
 28007 MADRID

Delegación en Cataluña:
 C/. Tarragona, 110 — Telf. 325 10 58
 08015 BARCELONA

DE VENTA EN EL CORTE INGLES
 Y TIENDAS ESPECIALIZADAS

(tm) Marca Registrada por el Grupo Indescomp.

LIBROS EN CASTELLANO PARA TU AMSTRAD



Manual de Referencia Basic para el Programador
La más autorizada y completa guía para programar en Locomotive Basic.
3.400.— Pts.



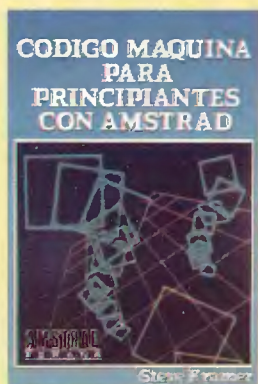
Juegos Sensacionales para AMSTRAD
Listados completos de 27 estupendos juegos de muy diversos estilos.
1.950.— Pts.



Programando con AMSTRAD
Fundamental para el usuario principiante.
Ameno y repleto de ejemplos.
2.400.— Pts.



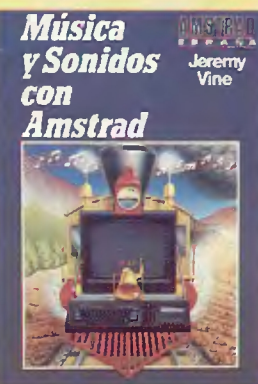
40 Juegos Educativos
Listados completos (matemáticas, geografía, música, etc.) para aprender divirtiéndose.
1.950.— Pts.



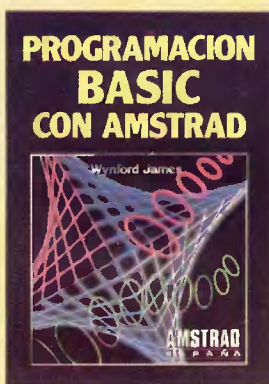
Código máquina para principiantes con AMSTRAD
Ideal para iniciarse en el código máquina del Z80 y en el sistema operativo del AMSTRAD.
2.100.— Pts.



Hacia la Inteligencia Artificial con AMSTRAD
Convierta su AMSTRAD en un compañero inteligente.
1.500.— Pts.



Música y Sonidos con AMSTRAD
Programa música y efectos sonoros y convierta su AMSTRAD en un sintetizador.
1.200.— Pts.



Programación de Basic con AMSTRAD.
Imprescindible para el principiante y eficaz herramienta para el programador avanzado.
2.100.— Pts.



Técnicas de Programación de Gráficos en el AMSTRAD
Este libro enseña a aprovechar las excelentes funciones gráficas del AMSTRAD, con múltiples ejemplos.
1.950.— Pts.



Curso Autodidáctico de Basic I y II
Un completo y estructurado Curso de Basic apoyado con numerosos ejemplos y acompañado de cassettes.
2.900.— Pts. cada volumen



indescorp
PUBLICACIONES

Avd. del Mediterráneo, 9
Telfs.: 433 45 48 — 433 48 76
28007 MADRID

Delegación en Cataluña:
C/. Tarragona, 110 — Telf. 325 10 58
08015 BARCELONA

DE VENTA EN EL CORTE INGLES
Y TIENDAS ESPECIALIZADAS

™ Marca Registrada por el Grupo Indescorp.

AMSTRADtm

"Lo increíble"

Confirmado por la prensa especializada

tu Micro



Micro, cassette y monitor en plena armonía. Su Basic es el más rápido de su categoría, superando al del Commodore, al del BBC e incluso al del Sinclair.

Computer Schau



Usuarios y técnicos lo confirman: se ofrece una relación precio/prestaciones que parece imposible.

Computer persönlich



Por un precio sorprendente se ofrece algo increíble. Un Basic superlativo.

micro



No hay en el mercado ningún ordenador en este nivel de precio que pueda enfrentarse a él.

C'T



¡Solución total a un precio fenomenal!

POPULAR Computing WEEKLY



Un ordenador personal extraordinario con unas enormes posibilidades como ordenador de gestión.

Personal Computer World



Su Basic es rápido, más rápido que casi todos los basics de 8 bits y que algunos de 16 bits.

micro bit



Su Basic se puede considerar impresionante... tiene unas características no usuales en microordenadores de su categoría.

COMPUTER CHOICE



El mejor Basic que he visto.

SCIENCE VIE MICRO

Se asedia a los distribuidores para conseguir un AMSTRAD. La demanda es desbordante.

micros



Calificado de "increíble", las pruebas realizadas así lo han confirmado en casi todos los aspectos... es un equipo con posibilidades fuera de lo común...



AMSTRADtm ***"Lo increíble"***